A Cooperative Framework for Molecular Biology Database Integration Using Image Object Selection

A thesis submitted to School of Computing Science, Middlesex University in partial fulfillment of the requirements for the degree of Ph.D

Nawaz Khan
January, 2004

ABSTRACT

The theme and the concept of 'Molecular Biology Database Integration' and the problems associated with this concept initiated the idea for this Ph.D research. The available technologies facilitate to analyse the data independently and discretely but it fails to integrate the data resources for more meaningful information. This along with the integration issues created the scope for this Ph.D research.

The research has reviewed the 'database interoperability' problems and it has suggested a framework for integrating the molecular biology databases. The framework has proposed to develop a cooperative environment to share information on the basis of common purpose for the molecular biology databases. The research has also reviewed other implementation and interoperability issues for laboratory based, dedicated and target specific database. The research has addressed the following issues:

- diversity of molecular biology databases schemas, schema constructs and schema implementation
- multi-database query using image object keying
- database integration technologies using context graph
- automated navigation among these databases

This thesis has introduced a new approach for database implementation. It has introduced an interoperable component database concept to initiate multidatabase query on gene mutation data. A number of data models have been proposed for gene mutation data which is the basis for integrating the target specific component database to be integrated with the federated information system. The proposed data models are: data models for genetic trait analysis, classification of gene mutation data, pathological lesion data and laboratory data. The main feature of this component database is non-overlapping attributes and it will follow non-redundant integration approach as explained in the thesis. This will be achieved by storing attributes which will not have the union or intersection of any attributes that exist in public domain molecular biology databases. Unlike data warehousing technique, this feature is quite unique and novel. The component database will be integrated with other biological data sources for sharing information in a cooperative environment. This involves

developing new tools. The thesis explains the role of these new tools which are: meta data extractor, mapping linker, query generator and result interpreter. These tools are used for a transparent integration without creating any global schema of the participating databases.

The thesis has also established the concept of image object keying for multidatabase query and it has proposed a relevant algorithm for matching protein spot in gel electrophoresis image. An object spot in gel electrophoresis image will initiate the query when it is selected by the user. It matches the selected spot with other similar spots in other resource databases. This image object keying method is an alternative to conventional multidatabase query which requires writing complex SQL scripts. This method also resolve the semantic conflicts that exist among molecular biology databases.

The research has proposed a new framework based on the context of the web data for interactions with different biological data resources. A formal description of the resource context is described in the thesis. The implementation of the context into Resource Document Framework (RDF) will be able to increase the interoperability by providing the description of the resources and the navigation plan for accessing the web based databases. A higher level construct is developed (*has, provide and access*) to implement the context into RDF for web interactions. The interactions within the resources are achieved by utilising an integration domain to extract the required information with a single instance and without writing any query scripts. The integration domain allows to navigate and to execute the query plan within the resource databases. An extractor module collects elements from different target webs and unify them as a whole object in a single page. The proposed framework is tested to find specific information *e.g.,* information on Alzheimer's disease, from public domain biology resources, such as, Protein Data Bank, Genome Data Bank, Online Mendalian Inheritance in Man and local database. Finally, the thesis proposes further propositions and plans for future work.

Acknowledgements

First, I would like to express my thanks to my Director of Studies, Dr. Shahedur Rahman for all the help, support and encouragement he has given me throughout this research work. He has guided me to stay on track and he has helped me to focus on a specific topic of this research. He provided constructive feedback and support to shape up this thesis. More specifically, this work could not have been done without his moral support and encouragement.

Thanks must also go to my second supervisor, Dr. Tony Stockman for his constructive feedback and encouragement. His valuable suggestions have enriched this thesis.

Thanks must also be given to others for their frank and generous comments in helping to make the research more coherent and cohesive at its early stage. They are: Dr. Chris from Birkbeck College, University of London and Professor T. G. Clarkson from King's College, University of London.

I would also like to thank the anonymous referees who reviewed the papers submitted for different conferences. The positive feedback from them encouraged me to carry out the work. I express my gratitude for giving the constructive feedbacks which helped me to reshape the research in the right direction.

I would like to thank Prof. Colin Tully, the Director of Research and the School of Computing Science for giving me the opportunity to do this Ph.D research. I would also like to thank Kerry Lane and Emma Warne for their cooperation through out the research period.

Finally, I would like to thank my wife, daughter, mother and sisters for their continuous encouragement, motivation and moral support through out this period.

Contents

List of Figures

List of Tables

List of Abbreviations

API	: Application Program Interface
ASN.1	: Abstract Syntax Notation 1
BLAST	: Basic Local Alignment Search Tool
CPL	: Conceptual Prototyping Language
DBs	: Databases
DBMS	: Database Management System
DDBJ	: DNA Database of Japan
DDL	: Data Definition Language
DML	: Data Manipulation Language
DOE	: Department of Energy, USA
DOM	: Document Object Model
EBI	: European Bioinformatics Institute
EER	: Extended Entity Relations
EMBL	: European Molecular Biology Laboratory
GDB	: Genome Data Bank
GenBank	: Gene Data Bank
GHT	: Generalised Hough Transform
GSDB	: Genome Sequence Database
HGMD	: Human Gene Mutation Data
HT	: Hough Transform
HTML	: Hypertext Markup Language
HTTP	: Hyper Text Transfer Protocol
IGD	: Integrated Genome Database
OMIM	: Online Mendelian Inheritance in Man
OPM	: Object Protocol Model
LoG	: Laplacian of Gaussian
MBD	: Molecular Biology Databases
PCR	: Polymerase Chain Reaction
PDB	: Protein Data Bank
NCBI	: National Centre for Biotechnology Information
RDBMS	: Relational Database Management Systems
RDF	: Resource Document Framework
ROI	: Region of Interest
SQL	: Structured Query Language
UID	: Unique Identifier
URI	: Unique Resource Identifier
URL	: Uniform Resource Location
WWW	: World Wide Web
W3C	: World Wide Web Consortium
XML	: Extensive Markup Language
XSLT	: Extensible Stylesheet Language for Transformation
2DWG	: Two Dimension Database of Web Gels
3D	: Three Dimension

Chapter 1

Introduction to Bioinformatics and Molecular Biology Databases

Chapter Objective

The chapter sets a background of the research topic. This chapter also explores widely used public domain databases, their structures, search and access mechanisms. The knowledge of these public domain databases which are presented here are the foundation of this research. Furthermore, the chapter highlights the thesis structure and the contribution of this research.

Chapter Contents

Chapter 1

Introduction to Bioinformatics and Molecular Biology Databases

Bio-informatics: bioinformatics is conceptualising biology in terms of molecules (in the sense of Physical chemistry) and applying "inforamtics techniques" (derived from disciplines such as applied maths, computer science and statistics) to understand and organise the information associated with these molecules, on a large scale. In short, bioinformatics is a management information system for molecular biology and has many practical applications.

Luscombe *et al.* (2001) : Oxford English Dictionary

1.1 Introduction

Molecular biology provides one of the most challenging application domains for database research. This is because of the rich variety of data which are easily available from genome sequences and protein structural data of full organisms, and the large quantities of data that are becoming available through modern experimental techniques. These pools of information makes bioinformatics an interesting and rewarding application area. It raises the scope to integrate and analyse these data which can lead to a better understanding of biological functions at all levels (Karp, 1996b and Bayat 2002).

The term bioinformatics refers to the application of computers in addressing biological problems, and it is most frequently used in relation to storing and searching genome sequence data and protein sequence data (Luscombe *et al.*, 2001 and Stevens *et al.*, 2001). In other word bioinformatics can be defined as information technology applied to the management and analysis of biological data. Computing power can be a useful assistant in automating, organising, and analysing this biological data. Bioinformatics covers areas like protein structure analysis, protein structure modelling, gene sequence matching, *etc.* (Bayat, 2002). Bioinformatics also covers other areas such as integration of heterogeneous molecular biology databases. This particular area is the main focus of this research.

This research is aiming to present a concept of how data can be extracted from the molecular biology databases in heterogeneous environment for more meaningful information. These databases store a variety of information and various

bioinformatics tools use these information to analyse problems raised by the biological research community. The following sections will overview molecular biology databases and its different aspects. The chapter presents the background of the research and identifies issues which are related to the research. This chapter introduces some existing molecular biology databases which will be used in the research and it describes the type of records and structures used by these molecular biology databases. The chapter also intends to introduce the bioinformatics tools and the users who are associated with these molecular biology databases. Finally, the chapter outlines the research scope and the motivation of this research.

1.2 Background

The data sets used in bioinformatics are essentially 'one-dimensional'. Genome sequences consist of letters which represent the sequence of the four bases in a nucleic acid chain. Protein sequences consist of letters which represent the sequence of amino acid residues in a protein chain. This wealth of information has only recently become available for further research in database applications. The major issues relating to this area are types of molecular biology databases, types of informatics projects and types of users. The following sections will be looking at these issues.

1.2.1 Molecular biology databases

There are several molecular biology databases which are considered to be major 'data resources' to store, maintain, update and to distribute these information (Wheeler *et al.* 2001). For example, one such data resource centre is the European Bioinformatics Institute (EBI) established in 1994. The major roles of the institute are to develop and to distribute the sequence databases. These databases store human chromosome sequence information for gene mapping. Basic features of the following major molecular biology databases (EMBL, GenBank, Genome Database, OMIM, PDB and HGMD) are discussed here.

EMBL

EMBL (European Molecular Biology Laboratory) is a laboratory that maintains Europe's primary nucleotide sequence data resource (Baxevanis, 2001). The EMBL Nucleotide Sequence Database is a comprehensive database of DNA (Deoxyribo Nucleic Acid) and the RNA (Ribo-Nucleic Acid) sequences which have

The EMBL Nucleotide Sequence Database is a comprehensive database of DNA (Deoxyribo Nucleic Acid) and the RNA (Ribo-Nucleic Acid) sequences which have been collected from scientific literature and patent applications. EMBL has been also collecting data directly from the researchers and the sequencing groups after peer review. It collaborates with GenBank (USA) and the DNA Database of Japan (DDBJ, Okayama, 1998). Table 1.1 describes the entities of the EMBL database.

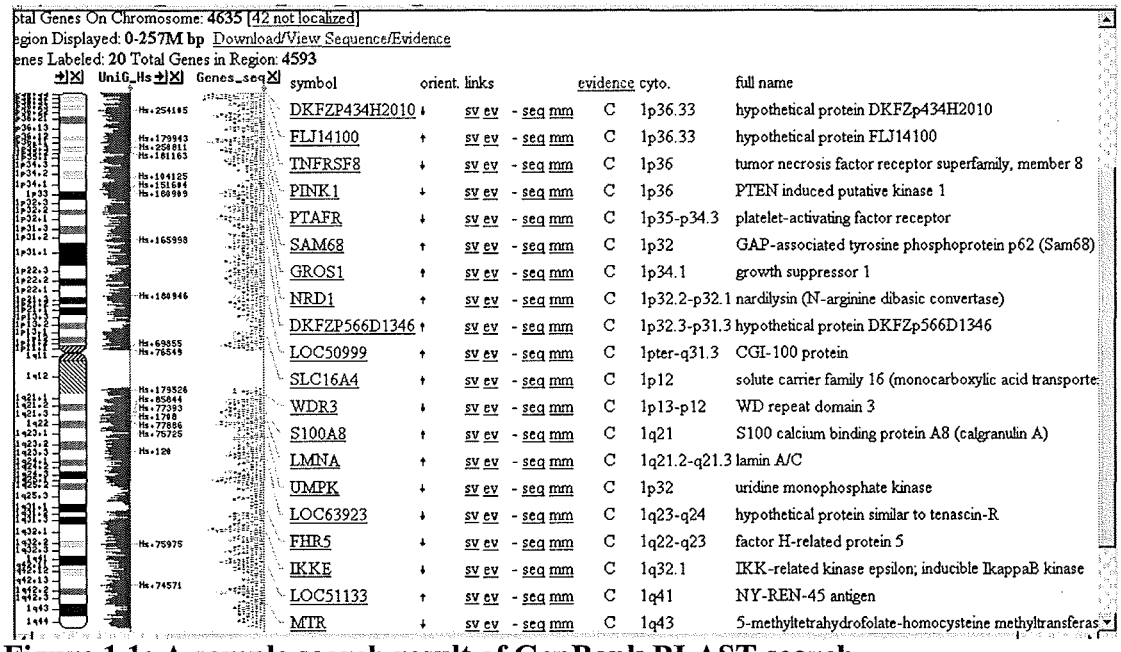
Table 1.1 Structure of the entities used in EMBL database

Attributes	Description	
ID	identification	(begins each entry; 1 per entry)
AC	accession number	(>=1 per entry)
SV	new sequence identifier	(>=1 per entry)
DT	date	(2 per entry)
DE	description	(>=1 per entry)
KW	keyword	(>=1 per entry)
OS	organism species	(>=1 per entry)
OC	organism classification	(>=1 per entry)
OG	organelle	(0 or 1 per entry)
RN	reference number	(>=1 per entry)
RC	reference comment	(>=0 per entry)
RP	reference positions	(>=1 per entry)
RX	reference cross-reference	(>=0 per entry)
RA	reference author(s)	(>=1 per entry)
RT	reference title	(>=1 per entry)
RL	reference location	(>=1 per entry)
DR	database cross-reference	(>=0 per entry)
FH	feature table header	(0 or 2 per entry)
FT	feature table data	(>=0 per entry)
CC	comments or notes	(>=0 per entry)
XX	spacer line	(many per entry)
SQ	sequence header	(1 per entry)
bb	(blanks) sequence data	(>=1 per entry)
//	termination line	(ends each entry; 1 per entry)

GenBank

GenBank (Benson *et al.,* 1993) maintains the database as a combination of flat files and relational databases containing Abstract Syntax Notation One (see Glossary). Each GenBank entry is assigned with a Unique Identifier by the NCBI (National Centre for Biotechnology Information). Since it emerged it has expanded somewhat in scope to include expressed sequence tag data, protein sequence data, three-dimensional protein structure, taxonomy and it links to the biomedical literature, MEDLINE (see Glossary). The database size approximately doubles every eighteen months. The average user of the database is not able to access the structure of the data

for query or for other functions, although complete snapshots of the database are available to export in a variety of formats, including abstract syntax notation one (ASN.1). The query mechanism provides WWW version which allows key searching of sequence and GenBank *UID* (unique identifier) searching through a static interface. Each GenBank entry includes a concise description of the sequence, *e.g.*, the scientific name and taxonomy of the source organism, and a table of features that identify coding regions and other sites of biological significance, such as transcription units (see Glossary), sites of mutations (see Glossary) or modifications, and repeats. Protein translations for coding regions are included in the feature table. Bibliographic references are included along with a link to the MEDLINE unique identifier for all published sequences. A sample search result of human *chromosome1* using GenBank BLAST (see Glossary) search is illustrated in Figure 1.1. The result shows the human *chromosome1* map. The record represents proteins associated with particular gene of *chromosome1*, for example, *PINK1* gene represents *PTEN* induced *putative kinase1*. Data values of the databases are linked with other databases using hyper link.


otal Genes On Chromosome: 4635 [42 not localized]
egion Displayed: 0-257M bp Download/View Sequence/Evidence
enes Labeled: 20 Total Genes in Region: 4593

symbol	orient.	links	evidence	cyto.	full name
DKFZP434H2010	↓	sv ev - seq mm	C	1p36.33	hypothetical protein DKFZp434H2010
FLJ14100	↑	sv ev - seq mm	C	1p36.33	hypothetical protein FLJ14100
TNFRSF8	↓	sv ev - seq mm	C	1p36	tumor necrosis factor receptor superfamily, member 8
PINK1	↓	sv ev - seq mm	C	1p36	PTEN induced putative kinase 1
PTAFR	↓	sv ev - seq mm	C	1p35-p34.3	platelet-activating factor receptor
SAM68	↑	sv ev - seq mm	C	1p32	GAP-associated tyrosine phosphoprotein p62 (Sam68)
GROS1	↑	sv ev - seq mm	C	1p34.1	growth suppressor 1
NRD1	↑	sv ev - seq mm	C	1p32.2-p32.1	nardilysin (N-arginine dibasic convertase)
DKFZP566D1346	↑	sv ev - seq mm	C	1p32.3-p31.3	hypothetical protein DKFZp566D1346
LOC50999	↑	sv ev - seq mm	C	1pter-q31.3	CGI-100 protein
SLC16A4	↑	sv ev - seq mm	C	1p12	solute carrier family 16 (monocarboxylic acid transporte
WDR3	↓	sv ev - seq mm	C	1p13-p12	WD repeat domain 3
S100A8	↑	sv ev - seq mm	C	1q21	S100 calcium binding protein A8 (calgranulin A)
LMNA	↑	sv ev - seq mm	C	1q21.2-q21.3	lamin A/C
UMPK	↓	sv ev - seq mm	C	1p32	uridine monophosphate kinase
LOC63923	↓	sv ev - seq mm	C	1q23-q24	hypothetical protein similar to tenascin-R
FHR5	↓	sv ev - seq mm	C	1q22-q23	factor H-related protein 5
IKKE	↓	sv ev - seq mm	C	1q32.1	IKK-related kinase epsilon; inducible IkappaB kinase
LOC51133	↑	sv ev - seq mm	C	1q41	NY-REN-45 antigen
MTR	↓	sv ev - seq mm	C	1q43	5-methyltetrahydrofolate-homocysteine methyltransferas

Figure 1.1: A sample search result of GenBank BLAST search

Genome Database (GDB)

GDB is built around a commercial relational DBMS SYBASE and it is modelled using standard Entity-Relationship techniques (Cuticchia *et al.*, 1993). There have

been difficulties in using this model to capture simpler map and to probe data because the data needed to draw a gene map is difficult to model in a relational database management systems. In order to improve data integrity and to simplify the programming for application writers, GDB provides a Database Access Toolkit. GDB is built around ten inter-linked data managers. Most users use a web interface to search the ten inter-linked data managers. Each manager keeps track of the links for one of the ten tables within GDB system with GenBank. Users are given only a very high-level view of the data at the time of searching and thus the users cannot easily make use of the information collected from the structure of GDB tables. Since, the structure of the tables are hidden from the users, the users are unable to do exploratory ad-hoc searching of the database using the present interfaces. Integration of the GDB database structure and On-line Mendelian Inheritance in Man (OMIM) has never been achieved (Fasman *et al.* 1996). OMIM database deals with genetic disorder diseases and it needs to have data correlation between GDB and OMIM data for analysing diseases due to genetic disorder. At present, GDB provides descriptions of the following types of objects:

- Regions of the human genome, including genes, clones, PCR markers, breakpoints, cytogenetic markers (see Glossary), fragile sites, syndromic regions, contigs and repeats.
- Maps of the human genome, including cytogenetic maps, linkage maps, radiation hybrid maps, content contig maps and integrated maps (see Glossary). These maps can be displayed graphically via the Web.
- Variations within the human genome including mutations, polymorphisms, and allele (see Glossary) frequency data.

GDB draws genetic maps based on researchers' information, *e.g.*, left flanking marker (see Glossary) and right flanking marker (see Glossary), however, it does not store the images of these genetic maps. For example, the following *APC* (*Adenomatous Polyposis Coli*) gene map (Figure 1.2) has been retrieved from GDB. The map shows *APC* gene location at the human chromosome. This map has been drawn based on left and right flanking marker values. However the image of the *APC* gene location can not be obtained from this map.

It is important to have the scope for correlation between gene map, Protein data bank (PDB) and OMIM need to be achieved in order to analyse diseased gene but this correlation can not be achieved in GDB database with its present data structure.

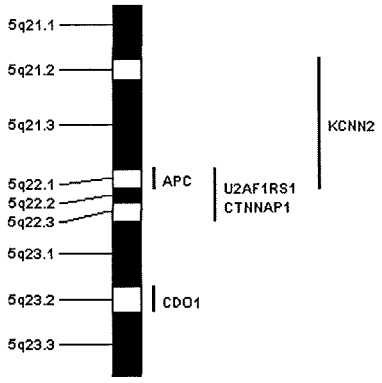


Figure 1.2 Gene map of *APC* gene (retrieved from GDB).

OMIM

On-line Mendelian Inheritance in Man (OMIM) is an electronic repository of information on the genetic basis of human diseases (McKusick, 1991). OMIM covers material on five disease areas based loosely on organs and systems. The phenotype (see Glossary) and genotype (see Glossary) entries contain textual data which are loosely structured as general descriptions, nomenclature, modes of inheritance, variations, gene structure, mapping, and numerous lesser categories. The full-text entries are converted to an ASN.1 structured format. The basic form of the database still remains difficult for any modification. Table 1.2 shows a list of OMIM database attributes which can be used for the search field, *e.g.*, search can be initiated for 'MIM number' field.

OMIM provides only textual information of the genetic disorder diseases, and it has not established any bi-directional links with other databases for variance analysis of diseases. It does not also store gene mutation information which are not associated directly with any diseases.

Table 1.2 OMIM database fields and its descriptions.

Search Field	Description	Qualifier
All Fields	Contains all terms from all searchable database fields in the database.	[ALL]
Allelic Variant	Describes a subset of disease-producing mutations.	[AV] or [VAR]
Chromosome	The chromosome onto which a gene or disorder has been mapped, as reported in the OMIM Gene Map.	[CH] or [CHR]
Clinical Synopsis	Clinical features of a disorder and the mode of inheritance (*e.g.*, autosomal dominant, autosomal recessive, x-linked).	[CS] or [CLIN]
Contributor	Contributor to an OMIM record. Names are in the format of lastname followed by one or more initials.	[AU] or [CTRB]
Creation Date	The date on which an OMIM record was created.	[CD] or [CDAT]
EC/RN Number	Number assigned by the Enzyme Commission or Chemical Abstract Service (CAS).	[EC] or [ECNO]
Editor	Editor of OMIM record.	[ED] or [EDTR]
Filter	Primarily used to retrieve subsets of records that contain crosslinks to other Entrez databases, and LinkOuts to external (non-Entrez) resources.	[FI] or [FILT]
Gene Map	Cytogenetic map location represented in the OMIM Gene Map.	[GM] or [MAP]
Gene Map Disorder	Text words appearing in the Disorder column of the OMIM Gene Map.	[DIS] or [DI]
Gene Name	The official gene symbol, and alternate gene symbols, associated with a record.	[GN] or [GENE]
MIM Number	**For information on the numbering system.**	**[ID] or [MIM]**
Modification date	Date on which the record was last modified.	[MD] or [MDAT]
Mod. History	All dates on which an OMIM record was updated.	[MDH] or [HIST]
Properties	An index containing various properties of OMIM records, identifying those which have attributes such as Allelic Variants, Clinical Synopsis, Mini-MIM condensed entry, or Gene Map locus.	[PR] or [PROP]
Reference	Contains author names and title words from the articles cited in an OMIM entry. Names are in the format of lastname followed by one or more initials.	[RE] or [REF]
Text Word	Contains terms from the main text-containing Section of a record.	[TXT] or [WORD]
Title Word	Words in title of an OMIM record. Includes words in the primary title, alternative titles, and included titles.	[TI] or [TITL]

Protein Data Bank (PDB)

Protein Data Bank (Bernstein *et al.,* 1977) is a collection of three dimensional protein structures which have been obtained by X-ray crystallography or nuclear magnetic resonance. The PDB is distributed in flat-file format, however data taken from the PDB for various projects have been reorganised using relational or object based system (Abola *et al.*, 1998). An example of PDB entry, its record format and structure are shown in Table 1.3, Table 1.4 and Table 1.5 respectively.

Table 1.3 Sample PDB entry

```
COMPND  MOL_ID: 1;
COMPND  2 MOLECULE: S-ADENOSYLMETHIONINE SYNTHETASE;
COMPND  3 CHAIN: A, B;
COMPND  4 SYNONYM: MAT, ATP\:L-METHIONINE S-ADENOSYLTRANSFERASE;
COMPND  5 EC: 2.5.1.6;
COMPND  6 ENGINEERED: YES;
COMPND  7 BIOLOGICAL_UNIT: TETRAMER;
COMPND  8 OTHER_DETAILS: TETRAGONAL MODIFICATION
```

Table 1.4 Record format of PDB database

COLUMNS	DATA TYPE	FIELD	DEFINITION
1 - 6	Record name	"COMPND"	
9 - 10	Continuation	continuation	Allows concatenation of multiple records.
11 - 70	Specification	compound	Description of the molecular list components.

Table 1.5 Detail structure of PDB database

TOKEN	VALUE DEFINITION
MOL_ID	Numbers each component.
MOLECULE	Name of the macromolecule.
CHAIN	Comma-separated list of chain identifier(s).
FRAGMENT	Specifies a domain or region of the molecule.
SYNONYM	Comma-separated list of synonyms for the MOLECULE.
EC	The Enzyme Commission number associated with the molecule.
ENGINEERED	Indicates that the molecule was produced using recombinant technology or by purely chemical synthesis.
MUTATION	Describes mutations from the wild type molecule.
BIOLOGICAL_UNIT	If the MOLECULE functions as part of a larger biological unit, the entire functional unit may be described.
OTHER_DETAILS	Additional comments.

An example of protein crystallographic information (see Glossary) search for *idcode:4HHB* using PDB is shown in Figure 1.3.

Title: **The crystal structure of human deoxyhaemoglobin at 1.74 A resolution.**
Compound: **Hemoglobin (Deoxy)**
Authors: **G. Fermi, M. F. Perutz**
Exp. Method: **X-ray Diffraction**
Classification: **Oxygen Transport**
Source: **not available**
Primary Citation: **Fermi, G., Perutz, M. F., Shaanan, B., Fourme, R.: The crystal structure of human deoxyhaemoglobin at 1.74 A resolution. *J Mol Biol* 175 *pp.* 159 (1984)**
[Medline]

Deposition Date: **07-Mar-1984** *Release Date:* **17-Jul-1984**

Resolution [Å]: **1.74** *R-Value:* **0.135**
Space Group: **P 21**
Unit Cell: *dim [Å]:* *a* **63.15** *b* **83.59** *c* **53.80**
angles [°]: *alpha* **90.00** *beta* **99.34** *gamma* **90.00**

Polymer Chains: **A, B, C, D** *Residues:* **574**
Atoms: **4779**
HET groups:

ID	Name	Formula
HEM	PROTOPORPHYRIN IX CONTAINING FE	$C_{34}H_{32}N_4O_4FE_1$
PO4	PHOSPHATE ION	O_4P_1

Other Versions: 2HHB, 3HHB

Figure 1.3. A crystallographic information search for *IDCODE:4HHB* using PDB

A crystallographic information for *idcode:4HHB* protein molecule is retrieved from PDB database. This crystallographic information is used for protein image modelling. Again, the correlation option between this protein image and GDB gene map for gene variance analysis is not available.

HGMD

Human Gene Mutation Database (HGMD) comprises published single basepair substitution in coding, regulatory and splicing relevant regions of human nuclear genes (see Glossary), as well as deletions, insertions, repeat expansions, combined micro insertions and deletions ('indels') (see Glossary) and complex rearrangements (Cooper *et al.*, 1999). It has reference to the first reported literature which has the following information: mutation, the associated disease states, the gene name, symbol and chromosomal location. All entries are allocated a unique accession number. By July 1999, HGMD contained approximately 18,200 lesions from a total of 895 human genes. When categorized by mutation type, the database included nearly 12,900 different single base pair substitutions, 3000 small (<=20bp) deletions, 1,100 small insertions, and 137 small indels. In addition, 1200 reports of gross gene deletions, insertions (>20bp) and complex rearrangements were also included (Krawczak *et al.*, 2000). In addition to data on the nature and location of mutations, HGMD also provides access to a collection of reference cDNA (see Glossary) sequences of human genes. Up to now, only mutations which are clearly causative for a particular inherited disorder have been logged into HGMD. Mutations are not necessarily disease-causing but may instead only be associated with one or more disease phenotypes (see Glossary) or particular disease linking which has not yet been established. Such variants are not included in HGMD. HGMD also does not store the indirect association of diseases which can not be determined by simple descriptive statistics or standardised information format. Complex mathematical modelling is necessary to determine the disease association with a particular mutated genes (Krawczak *et al.*, 2000). A list of attributes of HGMD database is shown in Table 1.6.

Table 1.6 Record structure of HGMD database

Gene Symbol
Gene Description
Nucleotide substitutions (missense / nonsense)
Nucleotide substitutions (splicing)
Nucleotide substitutions (regulatory)
Small deletions
Small insertions
Small indels
Gross deletions
Gross insertions & duplications
Complex rearrangements (including inversions)
Repeat variations

The previous Sections have described the major sources of genome and their functional data sets. The data structure of the individual databases reveals that each database has been constructed to serve a particular type of information to the scientific community. Combining all the related information can serve more meaningful information and it will also be able to contribute in genetic variance analysis for understanding the underlying mechanism of diseases. The major obstacle to integrate these databases is the heterogeneous nature of these databases. The heterogeneous features of the databases have been summarised in Table 1.7. The table summarises the major contents of the data resources. It also highlights the current structure and the major data types used in these molecular biology databases.

Table 1.7 Heterogeneous features of the major genome related databases

Database Name	Major Contents	Current Structure	DB Problem Areas	Primary Data types
GenBank	DNA/RNA Sequence Protein	Flat-file/ASN.1	Schema browsing, schema evolution, linking to other dbs	text, numeric
OMIM	Disease Phenotypes, Genotypes	Flat-file/ASN.1	Unstructured, linking to other dbs	Text
GDB	Genetic Map Linkage data	Relational	Schema expansion, complex objects, Linking to other dbs	Text, numeric
HGMDB	Sequence and Sequence Variants	Flat-file-application specific	Schema expansion, linking to other dbs	Text

1.2.2 Types of bioinformatics projects

Computer systems play essential roles in all aspects of genome research, starting from data acquisition and analysis to data management. The high-volume data intense bioinformatics projects require powerful computers and appropriately designed data-management systems (Kingsbury, 1993 and Blake, 1995).

The Report of the Invitational DOE workshop on Genome informatics identified three main components for genome related information projects. These components are data acquisition, data analysis and data-management system (Kingsbury, 1993). These issues are discussed in the following Sections.

1.2.2.1 Data-acquisition systems

The report emphasises that an appropriate and efficient data acquisition system is required for research laboratories which are generating large amounts of data. All the Genome centres and similar laboratories need to be supported by the strong local informatics group for acquiring data efficiently. This Section presents some examples of bioinformatics data acquisition systems which include inventory control system, sequence production software and visualisation software. A brief description of these systems are provided here:

Inventory Control Software

A major genome centre may require several hundreds of thousands of reagents, gels, and other materials. The automated inventory control systems are needed to track the use of these materials. The manual tracking will not only be possible but it will not be efficient either. These type of systems are now becoming essential for laboratory management.

Sequence Production Software

At present, machine learning and artificial intelligence are becoming major components for any computer systems that are used to analyse almost all aspects of sequence generation and assembly.

Visualization Software

Invariably most of the genomic data are presented as images and efficient tools are required to interpret these images in gel electrophoresis (Chapter 4), to read filters and to execute other steps on the critical path for genome analysis. Developing appropriate software for storing these images is the major component of any bioinforamtics project. The image data will enable the users to conduct

comprehensive analysis of morphological differences for genome and genome products using image-processing techniques.

1.2.2.2 Data-analysis systems

The functional understanding of Genome data cannot be revealed without the use of powerful computer systems (Bayat, 2002). But studying sequences, predicting protein structures, and comparing genomes to greater extent also need additional informatics tools (Letovsky, 1995), such as sequence analysis software, protein folding software, physical mapping and contig assembly software, genetic mapping software, comparative genomic tools and classification software.

1.2.2.3. Data management systems

Huge amount of functional genomic and phenotypic data are generating from the genome project (Paton *et al.* 2000 and Okayama *et al.,* 1998). At present, this data cannot be accommodated by traditional publishing methods (Krawczak *et al.* 2000). The current databases should be able to provide ready access to the data, find the essential data, interpret current experiments and to plan for future work. At present, electronic data management and data dissemination systems are increasingly becoming crucial components in bioinformatics (Bayat, 2002 and Krawczak *et al.* 2000). The nature of these databases range from highly specialised databases supporting local research projects (Khan *et al.* 2001a) to general databases that support the entire community (Robbins, 1994).

Local Databases

Local databases need to be targeted and dedicated. These databases should be flexible enough to handle specific local needs. The raw laboratory data and the local analysis of these data are required to be stored. Not only the local researchers should be able to get benefit from it but the large community should also have access to these databases for inter-laboratory data exchange or for data comparison. The integrated local databases should allow the flexibility of coping with the rapid changes in local experimental protocols and results. The technical issues of local databases should be addressed by the local users and this technology must be able to meet quickly the specific requirements defined by the local researchers (Davies *et al.,* 1997).

Collaborative Databases

Collaborative databases are conceptually between local and community systems (Kingsbury, 1993). For example, a set of collaborating researchers may work on any particular chromosome and these information need to be stored in a central

repository. This type of database includes both raw data and refined information, for example, *Chromosome19* Database. The implementation of a collaborative database requires developing distributed computing environment. The distributed computing environment can be established by interconnecting the computers either by using local high speed network or through the internet (Jain, 2002).

Community Databases

The community databases are shared resources. These databases are open to the entire research community and the majority of their users are located at remote sites. In most cases the community databases provide refined information instead of raw data. However, at present these databases are also attempting to make the raw data available. For example, GDB now provides raw data on which a published genetic map is based. There are several existing approaches to build community databases, for example, data warehousing, federated databases *etc.*, but these approaches need to be robust and it needs to meet the requirement of the entire community. These requirements lead to new challenges, such as, design consideration, appropriate approach and sufficient flexibility to address the need of different user communities. The successful design of community databases depend on the construction of component databases and these component databases should be flexible enough so that it can be fitted within the large biological science information infrastructure. This will allow different users to integrate the view from the component databases for greater understanding of the data (Khan and Rahman, 2002b). Moreover, community databases should recognise the data correlation among the multiple databases to demonstrate the biological interdependence and should provide support for integrated queries involving multiple databases. They should be also well documented for database integration and interoperation.

1.3 Molecular Biology Data: Current Problems and Approaches

It is apparent that the community databases should be integrated with each other in this era of comparative and functional genomics and proteomics research. It is evident by various research that not any single consolidated database will be able to serve the entire research community (Karp, 1995 and 1996ab). The advancement of web technologies initiated some degree of linking among the molecular biology databases. Many information resources are now available in web but they are not presented in a systematic and structured manner. The magnitude of biological web

data is increasing on a regular basis, but the data correlation between these web based information is almost impossible. To address this problem, linking biological web data based on their meaning has now become a major research area in Computer Science and Bioinformatics. Other approaches, such as, data warehousing (Schonbach *et al.* 2000 and Markowitz and Topaloglou 2001) and database federation for biological data integration emerged as an alternative to web linking (Kemp *et al.* 2000a), but these approaches have its own limitations as explained in Chapter 2. For example, these approaches do not allow local autonomy and schema transformation and they fail to address the 'global' issues that exist in molecular biology.

The present approaches in molecular biology database integration concentrate on building a 'Federated Information System' (Busse *et al.*, 2000) as a world-wide information solution. But to include the web data in 'Federation Information System' requires defining a web data model, developing correlation with the objects and navigating within the resources. The major challenge in achieving this goal is to build an effective framework and to focus on cooperative environment where the integration will be an automated process with or without minimum intervention from the users.

Next Section describes the research scope and motivation for this research.

1.4 Scope of the Research

Comprehensive studies of molecular biology data often involve exploring multiple molecular biology databases. To explore these databases efficiently requires dealing with the distribution of data among these databases. This involves synchronisation with the heterogeneity of the systems underlying these databases and the semantic (schema representation) heterogeneity of these databases. Karp (1995) said in this context:

> Molecular Biology Databases are created by such a diverse set of international groups that nobody has the power to legislate the standards at any single level of abstraction, much less at all of the existing levels of heterogeneity, such as the conceptualization, the data model or the query language

Many other researchers also highlighted this issue, for example, Davidson *et al.* (1995) said:

> Biomatrix calls for integration of data on a huge scale in both volume and complexity, significant technical challenges still remain for the smaller scale data integration problems that arise frequently within bioinformatics

Some of the researchers also highlighted the importance of clear criteria for characterising systems which are managing heterogenous databases. For example Markowitz (1995a) said:

> Molecular biology data are distributed among multiple databases. Although containing related data, these database are often isolated and are characterized by various degrees of heterogeneity. They usually represent different views of the scientific domains and are implemented using different data management system. Currently several systems support managing data in heterogeneous molecular biology database. Lack of clear criteria for characterizing such systems preclude comprehensive evaluation of these systems or determining their relationship in terms of shared goals and facilities.

These references also show that the integration of molecular biology databases is a major challenge in bioinformatics. The challenge ranges from schema modelling to developing collective tools for interoperability in molecular biology databases. The scope of these issues can be summarized as follows:

- schema development
- data modelling for resolving semantic conflicts
- looking at data submission and data creation issues
- resolving heterogeneous system conflicts
- developing and maintaining database and its autonomy
- developing integration theory
- application of integration theory in the context of molecular biology databases
- developing theory for multidatabase query
- application of multidatabase query concepts in the context of molecular biology databases
- data interpretation and visualisation
- developing and maintaining a cooperative framework for datbabase integration
- data comparison and data analysis for more meaningful information
- automated and dynamic navigation for data exploration
- developing other methods for multidatabase query using image object keying

It is clearly evident that 'biological data integration' gives rise to many aspects which need to be investigated and researched.

1.5 Aims and Objectives of this Research

The primary aims of this research are to understand the theory of gene database integration and to identify the challenges it poses toward the bioinformatics community.

'Database interoperability' in bioinformatics deals with both the technology and the bioinformatics community. By looking at both of these perspectives, the research aims to investigate the present technologies and to establish a relationship between these two domains. The technical complexity that exists for genome database integration has been reported in many literature. Both Markowitz (1995a, 1995b and 1996) and Karp (1995) highlighted the molecular biology database integration issues describing the heterogeneity of these databases. This research will look at these issues and it will investigate the existing methods adopted in this research area. The research will initially investigate the perspective of the existing theories and then it will contribute in formulating a more adaptable and versatile method. The major contribution of the research will be to incorporate this as a new approach for molecular biology database integration. Finally, the research will explore how to overcome the present difficulties using this new approach and it will establish if these difficulties lie within the present technology or within the user domain.

This research will have the following objectives to achieve the stated aims:

- To provide understanding of current database structures
- To provide understanding about users need to explore the molecular biology databases
- To introduce a new concept for database integration
- To present a method for database integration on the basis of this new concept
- To introduce a framework of cooperative environment for database integration on the basis of this method
- Incorporate a new image based query concept

1.6 Contribution of this Research

The main contribution of this research will be in the field of bioinformatics and 'Molecular Biology Database Integration' from the technological and users perspective. A new method for database integration will have a major impact in this research area and the effect will be for long term. The proposed framework for database integration as described in this thesis will also provide a major influence in the decision making process of developing collective tools for database integration.

This work will facilitate the interaction with practitioners, researchers and academics as a result of better interoperability between different molecular biology resources. This will lead to more activities and participating in the form of conferences, meetings, publishing and other research activities.

Finally, the research will provide an aid to overcome the difficulties encountered in resolving semantic conflicts and in using query scripts. The research will highlight the need to develop a cooperative environment for analysing gene mutation data and for understanding molecular mechanism of the diseases. The research will also highlight the fundamental and technological guidelines to develop future tools for interoperability within the molecular biology databases. The outcome of the research has been published throughout the research period and Table 1.8 shows the corresponding sections which have been used for the published papers.

Table 1.8 Research papers and the related chapters

Research papers	Based on sections
Khan, N., Rahman, S. and Stockman, T; (2004), Consolidation of web data for genetic variance analysis. Conference on Intelligent Systems for Molecular Biology and European Conference on Bioinformatics (submitted)	7.3.1, 7.3.2, 7.4.1 7.4.2, 7.5, 7.7
Khan, N, Rahman, S and Stockman, T; (2003) Integration of biological resources using image object keying. 16th IEEE Computer Based Medical System. IEEE Press	7.2, 7.4, 7.5, 7.6, 7.7, 5.4
Khan, N and Rahman, S., (2003) A new approach to detect similar proteins for 2D gel images. IEEE Conference on Bioinformatics and Bioengineering.	5.2, 5.3, 5.4
Khan, N, Stockman, T. and, Rahman, S, (2002) A cooperative environment for genetic variance analysis using component database for database integration. 15th IEEE symposium on Computer Based Medical Application. Slovenia, 2002. Proc. IEEE computer society press.	7.2.1, 7.2.2, 5.3.4
Khan, N. and Rahman, S (2002) Object modelling of gene mutation data for variance analysis. 6th world conference on systemics, cybernatics and informatics (SCI, 2002); SCI in Medical and Biology session; Florida, USA Proc. International Institute of Information Systems.	3.3.1, 3.3.2
Khan, N. and Rahman, S.; (2001) A conceptual object modelling of gene mutation data. 2001, Proc. German Conference of Bioinformatics, GCB01. Germany.	3.4.2 and 3.4.3
Khan, N., Rahman, S and Clarkson, G. T.; (2001) An approach to develop human gene disorder database for intelligent variance analysis of genes and its products. 2001, 12th International workshop on Database and Expert System Germany, Munich. Proc. IEEE Computer society press.	3.2.1, 3.2.2, 3.4.3 7.2.1

1.7 Outline of Other Chapters

Chapter 2 describes the background of the research area. This chapter highlights other ideas from different researchers regarding 'molecular biology database interoperability'. The chapter also reviews and investigates other current projects in this area. The outcome of this review leads to formulate the scope for this research.

Chapter 3 sets out the approaches adopted to overcome the existing problems in molecular biology database integration. A formalism of framework is presented in chapter 3. The chapter continues with the following structure:

- identifying the difficulties
- discussing the issues to overcome such difficulties
- proposing an approach for improvements and describing a gene mutation data model
- summary of the chapter

The Chapter 4 discusses the approach used in selecting image object to initiate the integration process. Widely used gel electrophoresis protein images are used to implement this new approach. The approach has looked at different image processing and geometric methods for selecting an object. It has also looked at the matching algorithms. The research highlights the present needs and limitations. This chapter also builds an argument that initiating query by image object selection can resolve the semantic conflicts in molecular biology database search.

Chapter 5 outlines the proposed approach for image object selection and matching. An empirical evidence is presented to demonstrate the approach.

Chapter 6 introduces the present database integration methodologies. It highlights the drawbacks of the present methodologies and builds a formal description of context for integration.

Chapter 7 discusses a formal description of context graph approach. This approach defines the semantics of web data and it also describes an integration domain for successful context based integration. The chapter analyses the navigational approach for locating data resources and collecting the web data. The chapter ends with an example to demonstrate the use of context graph model.

Chapter 8 discusses the approaches used in the previous chapters. It outlines the advantages of the approach and the contributions that it will make in the domain of molecular biology. The chapter also highlights the scope for future research work.

A list of references is provided at the end of this thesis. The references are sorted by the last name of the authors. Author's last name and year of publication are used as a reference within the text of the thesis. Glossary has been attached as Appendix A. It provides the description of bioinformatics terms used in this thesis. Published papers have been appended in Appendix B. Appendix C lists the outline of the vocabularies that are used in RDF. It also provides the Document Object Model Interface definitions for HTML. Appendix D provides the source images that are used in this research.

1.8 Summary

This chapter has looked at different issues of molecular biological databases. The chapter has started by describing the concept of 'bioinformatics'. The term 'bioinformatics' has been defined as the information technology applied to effective management and to analysis of biological data. The biological data are scattered over a number of databases, *e.g.*, EMBL, GenBank and GDB. These databases stored and maintained biological raw data collected from different laboratories and the databases are considered as major sources of biological information. However, these biological sources are developed by specific international groups and different legislative approaches have been applied to these databases for data submission and data exchange. Because of this, these molecular biology databases exist in different size, format and models. Accessing these databases are restricted to the level of users and organisations. This also gives rise to heterogeneity issue and the heterogeneity of databases is presenting a major obstacle to correlate the data among these databases. So, it is essential to integrate these databases in a coherent fashion for any meaningful information and analysis. The databases described in this chapter reveal that each of the databases is very useful source of information within a particular domain. For example, OMIM deals with disease phenotype data, on the other hand HGMD database deals with gene mutation data which has direct correlation with diseases. Most comprehensive database is GenBank which provides information about gene map with coding regions, transcription units *etc.* But GDB also provides the same type of information. Although GenBank and GDB are providing the same type of information, these two databases have different architectures and models to represent their data. All these databases provide a fixed query and do not support any comparative analysis of the data. These databases can be accessed by using

conventional methods, *e.g.* by using hyperlink where each transaction is discrete and stateless. It does not allow multidatabase query for more meaningful information.

This chapter has also looked at different bioinformatics research works. Many bioinformatics research have emerged on the basis of these databases. Sequence analysis, genetic mapping and comparative genomic tools are the most promising fields in this area of research. The success of genomic projects depends on how quickly and effectively challenging questions can be answered and addressed. Each of these databases is holding a portion of the information which can not be integrated due to their heterogeneity. More specifically, biological data interdependence that has been stored in multiple databases needs to be recognised to support integrated query. This issue is throwing a challenge to database interoperation. Although the collaborative or community database approach as described here is the basis of database integration, the extent of interoperability and extraction of data for more meaningful analysis from these databases are still considered to be a major part of an ongoing research.

Database maintenance and types of users are equally a major concern for human genome projects. As well as archiving the databases, the need for dynamic analytical tools to support different community of the database users is essential. It is hard to believe that any one particular biological source will be able to serve all the bioinformatics community, rather autonomous and decentralised database with dynamic tools can act as component to the community database. The chapter has also introduced a number of areas of concern within this research domain. It has highlighted the aim and objectives of this research. Finally, the chapter outlines the research contributions in this area.

The next chapter will look at the issues concerning the molecular biology database diversity in the context of global schema development and semantics of data. It will focus on the approaches adopted by different researchers for integration of the databases for more meaningful information. The chapter will review the current works in this research area and it will identify the probable research issues and questions.

Chapter 2

Literature Review and Research Issues

Chapter Objective

Molecular biology databases presently exist in different formats, sizes and in different data structures. These create a new challenge to make these databases interoperable. Currently, hyperlink text is attempting to link these databases but it is unable to resolve semantic conflicts of the data. This chapter examines the diversity of molecular biology database in terms of data modelling, data storing and data semantics. The heterogeneous nature of the molecular biology databases is throwing a major challenge to database integration. These challenges are examined in this chapter and the possible approaches to resolve these challenges are investigated. The chapter critically reviews the present approaches and it sets out the research questions for this Ph.D research.

Chapter Contents

Chapter 2

Literature Review and Research Issues

.....the certain core areas need greater research effort if the field is to progress. One of these is in the area of integration and standard specification. Different databases, data types and data structure hamper knowledge discovery and data mining.

Rodrigo (2002), Editorial Foreword, Applied Bioinformatics, vol:1, no:1

2.1 Introduction

This chapter looks into the diversity of molecular biology data modelling. The chapter goes through the literature review and identifies the research themes. Section 2.2 introduces the present approaches adopted for molecular biology database integration. The chapter critically reviews these approaches and determines how these approaches are affecting molecular biology database integration for more meaningful information. It also outlines the existing concepts on designing molecular biology database schemas. A mathematical representation of the basic schema construction has been described to explain this concept. Markowitz (1995) developed this schema model. The Section ends by proposing additional suggestions on this model.

Section 2.3 reviews the data warehousing concept (Sean, 1994) and its use for molecular biology database integration. The section also establishes that how this concept fails to integrate molecular biology databases for sharing information.

Section 2.4 identifies the problems of integrating biological data sources. It looks through the criteria proposed by different researchers (Markowitz, 1995c, Ritter, 1994 and Letovsky, 1995) for integrating molecular biology databases.

Finally, Section 2.5 raises questions and research issues. The Section then presents the limitations of some of the existing research projects. It then summarises the chapter.

2.2 Heterogeneity of Molecular Biology Data Modelling

Data which are of interest to molecular biologists are distributed over numerous heterogeneous molecular biology databases (MBDs). These MBDs display

heterogeneity at various levels, for example, they are implemented using different systems, such as structured files or database management systems (DBMSs). They are based on different view of the molecular biology domain, or they contain different conflicting data. Furthermore, each MBD represents only some part of the molecular biology domain and it is often designed to address only certain queries or applications. The data in a MBD are structured according to a schema specified in a data definition language (DDL) and they are manipulated using operations specified in the data manipulation language (DML). These languages are based on a data model that defines the semantics of their constructs and operations. The implication of this is that one particular database is a major source of a very specific raw biological data, but it is restricted to a particular scientific domain. Biological researchers rarely explore multiple databases for relevant information since there is no interoperability as such within the molecular biology databases. The main reason for this is their extended degree of heterogeneity. The heterogeneity of molecular biology databases can be seen as diversity of global schemas and views, diversity of data exploration and diversity of molecular biology database schemas (Markowitz *et al.,* 1996).

2.2.1 Data exploration from molecular biology databases

Exploring multiple MBDs entails coping with the distribution of data among these MBDs, with the heterogeneity of the systems underlying these MBDs, and with the semantic (schema representation) heterogeneity of the MBDs.

Strategies for managing heterogeneous MBDs can be grouped into two main categories (Bright *et al.,* 1992; Sheth and Larson 1990; Sheth and Kashyap, 1992 and Busse, 2000).

1. *Consolidation* strategies - entail replacing heterogeneous MBDs with a single homogeneous MBD formed by physically integrating the components of MBDs, or reorganising MBDs using a common DDL or DBMS.

2. *Federation* strategies - allow access to multiple heterogeneous MBDs, while the component MBDs preserve their autonomy, i.e., their local definitions, applications, and policy of exchanging data with other MBDs. Federation strategies include:

1. incorporating in MBDs references (links) to elements in other MBDs, or constructing MBDs consisting of such links
2. organising MBDs into loosely-coupled multidatabase systems and
3. constructing data warehouses.

Karp (1995) suggested that heterogeneous MBDs can be connected via hypertext links on the web at the level of individual data items relationship. However, Kemper (2001) argued that although integration of data item links using hypertext links between MBDs do not require or comply with schema correlation across MBDs, it must comply with the hyper query for data retrieval. It is also important to point out that data retrieval in hyper linked systems is limited in selecting an initial data item within one MBD but it then needs to be followed up by consecutive hyperlinks to reach the final data items. Numerous attempts have been made to integrate the molecular biology databases using hyper links and to automate the process of link extraction from the MBDs. For example, systems such as Sequence Retrieval Systems (SRS) (Etzold *et al.* 1993; Etzold *et al.*, 1996) and LinkDB (Goto *et al.*, 1995) extract existing link information from (usually flat file) MBDs, and construct indexes for both direct and reverse links allowing fast access to these MBDs. Although these systems attempted to resolve heterogeneity issues, they only provide simple index and key match retrieval, and they also lack the ability to support full query facilities (Markowitz, 1995c). For example, they cannot identify similar proteins which are responsible for different types of diseases.

Querying multidatabase systems which are collections of loosely coupled MBDs and which are not integrated using a global schema, involves constructing queries over subset databases of the whole federated molecular biology databases (Von, 2000). In this approach a query explicitly refers to the elements of each MBD involved. Chen *et al.* (1995a) proposed the Object-Protocol Model (OPM) tool-based strategy for querying component MBDs of multidatabase systems using a common query language. They further claimed that OPM has the capability to describe and to query the participating component MBDs using a common data model. They attempted to establish the argument that common query language approach does not require the component MBDs to be represented using a common DDL or data model. However, the users are required to have some knowledge regarding the structure of the MBDs for their query. The major drawback in this approach is that it requires all participating MBDs to have a view defined in a common DDL so that the users can

examine and query component MBDs in the context of the same data model. Chen *et al.* (1995ab and 1997) argued that the OPM model supports multidatabase query languages that allows specifying complex query conditions across MBDs which is not possible in hyper text linking approach. In OPM model, a query translator is needed for: (i) translating queries expressed in the multidatabase query language to query targeting component MBDs, and (ii) for optimising these queries.

Like OPM model, the data warehousing approach entails developing a global schema (view) of the component MBDs. Markowitz and Toponloglou (2001), Cornell *et al.* (2001) and Schonback (2000) applied the data warehousing approach in Bioinformatics. They expressed the MBDs in a common DDL and the discrepancies between these definitions are resolved prior to their integrating into the global schema. In this approach data from component MBDs are transformed in order to comply with the global schema and the data is then loaded into a central data repository. The Genome Information Management Systems (GIMS) (Cornell *et al.*, 2001) and Genome Topographer (GT) (Cozza *et al.*, 1994) are examples of data warehouses, where GIMS is developed with the Object Database POET system and GT is developed with the Gemstone commercial object-oriented DBMS. The query facilities of data warehouses are provided by the underlying system (*e.g.*, POET Object Database System), and the query processing is local to the warehouse. However, Kemp *et al.* (2000) and Kemper (2001) highlighted that constructing data warehouses require costly initial integration of the component MBDs. The data warehouse also requires to be synchronised frequently with the component MBDs in order to capture the evolution of their schemas. Moreover, data warehouses need to be updated on a regular basis in order to reflect the updates of the component MBDs.

2.2.2. Diversity of global schemas and views

At present diversification in molecular biology data model and its semantics are affecting the detail examination of the data in MBDs. In Bioinformatics domain, frequent access to the resources and high interconnectivity are the primary concerns for the efficient processing of data. Kashyap and Sheth, (1996) raised the issue of identifying the objects that are semantically related to each other and resolving the semantic differences among these objects. Existing systems for exploring heterogeneous MBDs do not address the problem of understanding the semantics of component MBDs. For example, systems that support links between MBDs do not

provide any information regarding the structure or semantics of the linked MBDs. Some multidatabase systems require users to know the structure (schemas) of component MBDs without providing them any support for this. The data warehousing concept (Schonbach, 2000 and Markowitz and Topaloglou, 2001) attempted to address the issues of database interoperability by utilising a single unified view. However, users are not aware of component MBDs which does not provide flexibility in terms of information focusing and correlating component across multiple databases. In order to correlate the multiple resources or to resolve the semantic differences of the data, the context of data comparison and the type of data abstraction at the semantic level need to be described. But semantic description and annotation at this level are hardly implemented in current widely used schemas. For example, IGD, GT, and Entrez are based on global schemas (views) of their component MBDs. These are expressed in ACeDB DDL for IGD and Gemstone DDL for GT and they are not well documented to describe the structure of the GT data warehouse. Providing the description of object relationships and correlation between the resources are not widely used practice in this research area.

Schonbach *et al.* (2000), emphasised on dimensional modelling in bioinformatics for data warehousing. By dimensional modelling he referred to the process which visualises and conceptualises the model for data handling and which helps in overcoming the gap between biological and computational requirements. However, the global schemas of systems such as GT, IGD, and Entrez are not based on dimensional modelling as described by Schonbach *et al.* (2000), instead they are the result of independent schema design processes based on the domain knowledge underlying the component MBDs. Most of the present global schemas do not represent `consensus' schemas. For example, GT, IGD, and Entrez were developed locally by small groups which did not take into account the 'consensus' approach and this affected the quality of design and implementation issues (Markowitz, 1995a). In order to reduce the complexity of schema design and to make global schemas general, developers usually design these schemas using `generic' classes and/or attributes which may not be applicable to individual component MBDs and which may not fully capture the semantics of the data (Markowitz, 1995b). Furthermore, the design of the global schemas of such integrated databases are expressed in system-dependent DDLs. So it shows potential risk of not reflecting domain modelling requirements and it may not be able to capture all the information of each component MBD. The

synchronisation of the data semantics with other data models needs to be correlated for effective interopeation. In this regard, Kashyap and Sheth (1996) suggested, abstractions or mappings between the objects for comparison and which are necessary to capture the semantic similarities for schema interoperability.

At present MBDs global schema constructs suffer from naming conflicts and inconsistencies in detecting identical entities of interest which are represented in different format and name. For example, the same concept can be represented in different schemas by using synonyms, alternative terminology, or different data structures in molecular biology. Both Gene Data Bank (GDB) and Gene Sequence Data Bank (GSDB) stores the genotypic details of gene but they use conflicting terms to represent the same object. For instance, in GDB a gene is represented as *gene* to represent a class of *gene*, whereas GSDB uses the term *genesequence*. Another could simply represent *locus* by directly using their sequence data. Markowitz and Ritter (1995) highlighted further examples of Homonyms and Domain conflicts. Homonyms can cause naming conflicts in a heterogeneous MBD environment. Domain conflicts can be caused by storing similar values using different units or formats in different MBDs, or from conflicting data coming from different experiments or experimental techniques. For example, an 'increase' can be represented by the word 'increase' or it can also be represented as '+' in the database. Other causes of conflicts include different ways of representing incomplete information (*e.g.*, the meaning of nulls) and identifying objects in different ways in MBDs.

As an attempt to resolve schema conflicts, many researchers have proposed different approaches which range from simple renaming to resolve naming conflicts to schema restructuring and resolving structural dissimilarities. Not much research has been carried out in the area of automation to resolve structural dissimilarities or naming conflicts in bioinformatics. This is mainly because of the complexity and high variable nature of data models. In many cases, a heterogeneous MBD system depends on users' intervention to resolve the semantic conflicts, for example in IGD, Entrez *etc.* If the semantic conflicts are left to the users then it needs to be standardised so that different communities can utilise the meaning of the data and the description of the structure for integration. However it has not been achieved yet simply because different communities use their own explanation and meaning for data conceptualisation and integration. So, the resolution of semantic difference needs to be automatic, dynamic and transparent to the users.

Kutsche and Sunbul (1999) have attempted to develop a reference architectural model for automating the integration process. They proposed a model with four layers which are Foundation, Integration by wrappers, Integration by mediators and application. The model introduced the concept of 'meta information' to express model correspondences and to include information modelling on different abstraction level. In their approach they defined the meta data to support the reusability and the documentation of the resources. Although, their approach is applied in Software Engineering, but it can be extended to heterogeneous distributed computing for bioinformatics. Shaker *et al.* (2002) described semantic mapping rules for providing mapping between heterogeneous databases and a mediated schema. In their approach, a meta wrapper performs the semantic transformation for each data source from its heterogeneous schema to the mediated schema. The bi-directional rules are not part of the meta wrapper, instead, they are stored in a knowledge base. The major drawback of this approach is the continuous update of the bi-directional rules in order to cope with the unstable data models. Thus, the solution of semantic conflicts does not lie in creating meta data or in developing a knowledge base rules, but it lies in developing a framework which can be extended and exploited in multi-agent architecture.

2.2.3. Diversity of data exploration

Markowitz and Topaloglou (2001), Schonbach *et al.,* (2000) and Cornell *et al.* (2001) suggested data warehousing approach to explore MBDs. In data warehousing approach, exploring MBDs require constructing schemas or views for data localisation. In this approach data are converted from the format of component MBDs to that of the data warehouse and then the data are loaded into the data warehouse. Miller *et al.* (1993) highlighted the necessity for preserving the *information capacity* of component MBDs when transforming one schema to another. The discrepancies between the *information capacities* of the views employed for exploring heterogeneous MBDs and the underlying component of MBDs can be a major source of confusion and it can lead to erroneous data integration.

Researchers make queries using SQL scripts for exploring biological data residing at different biological resources. The advent of web based data integration leads to the development of query form interfaces for biological query, for example *Entrez* and *WebEntrez*. However, the extent of facilities provided by these interfaces

for exploring heterogeneous MBDs have not been properly characterised (Markowitz *et al.,* 1996 and Markowitz and Topaloglou, 2001). Users interact directly with a Web query interface, *e.g. Entrez*, and they may not be aware of the existence of a global schema. It is not possible for the users to detect if there are any conflicts in component MBDs and to provide a solution to resolve the conflicts. For example, both Genome Database (GDB) and Genome Sequence Database (GSDB) have a *Gene* class. In GSDB 2.0, *genes* are considered to be a kind of *Feature* which are characterised by gene names and which are linked to the external databases. But in GDB *genes* are represented by objects of class *Gene* and they are characterised by information that includes the reason why a genomic region is considered a gene and these are linked to gene families to which the gene belongs to. Sequences are represented in GSDB by objects of class *Sequence.* Sequence data includes the actual sequence, sequence length, and the information on the source of the sequence. Sequence information in GDB is represented by objects of class *SequenceLink. SequenceLink* refers to the sequences in GSDB database. Both GDB and GSDB contain classes representing products. In GDB, products are limited to gene products, while in GSDB a product can be associated with any feature. So, query capabilities provided by these forms are limited and they are based on keyword matching. These interfaces do not provide the support for automated navigation to the resources and these are restricted to the query capabilities that are provided by the underlying system. For example, IGD is only capable of executing the queries that are supported by the ACeDB query language. ACeDB lacks many features for complex queries, *e.g.*, deductive capabilities (see Glossary), dynamic selection of resources and deriving context from the query. In contrast there are many powerful programming languages such as CPL which allows users to submit very complex queries. However, it is difficult to imagine a biologist will be able to write a complex query using such a complex language like CPL.

In molecular biology domain, web has become a very crucial information repository for exchanging and disseminating information. Many approaches are emerging to execute queries on web data. Sahuguet and Azavant (2001) in their approach introduced a middleware architecture using web wrappers to interact with the web sources, *e.g.* HTML documents and it then utilised a mediator to export a unified view of data to the users. This approach, known as the World Wide Web Wrapper Factory (W4F) parses the HTML into a well-formed document and it then

mapped in a DOM tree. A set of extraction rules are employed for navigation along the tree and it then specifies which piece of information that needs to be collected. The model is well accepted in wrapper generation case for structured and well organised data, such as small scale business data. But this approach needs to be extended to cope with bioinformatics data (Geihs, 2001). Because, W4F is based on the extraction rules and these extraction rules need to be rewritten as well as additional semantics need to be included in the wrapper.

2.2.4 Diversity of molecular biology database schemas

Some of the major databases have been analysed by Markowitz and Ritter (1995) and Davidson *et al.* (1999). It reveals that the basic construct of database schemas can vary quite widely depending on the types of definitions being used in their schema design. The variations were best explained in the second meeting on Interconnecting Molecular Biology Databases.

Markowitz (1995b) in this meeting described the basic construct of database schemas in terms of objects using attributes which are classified in classes. The objects of the classes can be reffered by a specific *identifier*. An attribute of a class can be single-valued, set-valued, or list-valued. In biological databases, the attribute in most of the cases consists of several component attributes, where each component attribute is associated with a single class of values or a union of classes. The attributes in molecular biology databases can be categorised as (i) primitive or (ii) abstract. The primitive attribute takes values from a controlled class of enumerated atomic values or ranges of values, or a class of atomic values of predefined data types. On the other hand, an *abstract* attribute takes values from a class or union of classes of objects. This implies that the attributes can be derived from other attributes of any class in molecular biology. In some cases, for example, to represent the complex data in biology, it is also necessary to characterise the attributes by inverse constraints for cross-referencing the object. For example, if an attribute A of class O_i is defined as the inverse of abstract attribute B of class O_j, then for every instance x of O_i and every instance y of O_j, if y is a value of $A(x)$ then x is also a value of $B(y)$. Also some derived attributes have been used for biological data modelling. Derived attributes have values that are derived from the values of other attributes using derivation expressions, such as arithmetic expressions, aggregate functions and attribute composition. For example, An attribute A of a class O_i, can be defined as a composition of other attributes by associating it with one path or a union of paths in

the following form: $B_1[O_{i1}]\ B_2\ [O_{i2}]...B_n[O_{in}]$, where each $O_{ik}(n \geq k \geq 1)$ denotes a class, and each $B_k(n \geq k \geq 1)$ denotes an attribute associated with $O_{i(k-1)}(O_{i0}=O_i)$ and takes values from value class that either includes or is a superclass of O_{ik}.

Markowitz (1995c) also reported that the biological database schemas contain two major types of derived (virtual) classes: (i) derived subclasses of one or several object classes, and (ii) derived superclasses of several classes. For example, a derived subclass, *Os*, of one or intersection of several object classes, $O_1,...O_m$, $(m \geq 1)$, and/or associated with a condition consists of the subset of objects that belong to the intersection of O_i and classes $m \geq i \geq 1$, and satisfy the associated condition. A derived superclass, O_g, of object classes $O_1,....,O_m$ $(m \geq 2)$, consists of the union of objects belonging to these classes. So, it can be concluded that the diversity and variation exist in schema design because the theoretical definitions are not unique in nature.

This complex, interconnecting attribute description make the schemas to dependent on each other which lead to the high risk of data inconsistency. Although, the practice of data modelling takes the approach of using Object Oriented model for data description which supports the decision of first Meeting on Interconnection of Molecular Biology Databases (Karp, 1994), but it leads to the schema integration challenge without data redundancy. Individual constructs are preferable to reduce the dependency of attributes to other schema attributes. The use of derived constructs makes the databases excessively complex and it results in redundant attributes and values. Some researchers argued to develop a standard data model for MBDs to facilitate MBD interoperation but standardisation could have a negative effect on MBD modelling. It is evident from the above discussion that the derived object constructs are not appropriate for modelling molecular biology data and hence additional linking with available information are needed for better modelling.

It is agreed by the researchers that the current molecular biology databases can not be modelled using structured data model. Instead, the model should be flexible enough to cope with the unstructured or semi-structured data. This implies that an extended independent construct with more semantics needs to be introduced. An extended and flexible model is needed to interconnect the diversified resources without affecting its local autonomy. The interoperability of this model can be achieved by introducing intermediate multi-agents which will interact with other MBDs within a cooperative framework (Khan *et al.*, 2001).

2.2.5 Modelling of genomic data

Bioinformatics projects have developed their own data models to accommodate data. Number of projects have been working on bioinformatics data modelling to achieve better storing and retrieving of data from different scientific domain. For example, Paton *et al.* (2000) designed a conceptual schema for functional genomic data using Unified Modelling Language (Booch *et al.*, 1999) and Xie *et al.* (2000) designed a relational data model for storing biological sequences. Biological data models are in constant rearrangement because of its dynamic nature and its constant requirements from the users. At the same time the schema design also needs to be redesigned at the same pace as the demand grows.

The overlapping data modelling problems still exist although many data models are emerging in bioinformatics. The data model needs to refer to other data models for sharing information. It is not desirable or even possible that a single data model will take control of the whole federation of databases. On the contrary, it is feasible to design a local data model which will be consistent and interoperable with other data models.

2.3 Approach to Integrate Heterogeneous Databases

Schema conversion is a key component of heterogeneous database systems. It underlines the construction of global views of multiple databases expressed in some common data model or local views of multiple databases which are expressed in the local data model of each component database. For example, the IGD system involves converting relational schemas into ACeDB schemas. It then integrates these schemas into a global ACeDB schema. However, this method of integration deals with generic attributes to serve a specific type of query (Schonbach *et al.*, 2000) which leads to the replication of data.

Schema conversion needs a measure of the *information capacity* (Miller *et al.*, 1993) preserved by the conversion. Thus, converting a schema S_1 in data model M_1 to schema S_2 in data model M_2 (M_1 and M_2 can be same or different data models), involves defining a data mapping (converter) that can transform instances of S_1 into instances of S_2. Informally, S_2 is said to preserve the information capacity of S_1 if the data mapping associated with the conversion of S_1 into S_2 transforms instances of S_1 into instances of S_2 without loss of information.

The following criteria are proposed (Miller *et al.*, 1993) for any schema conversion which will preserve the information capacity.

Let S_1 be a schema in data model M_1 and let S_2 be a schema in data model M_2. S_2 preserves the information capacity of S_1 if:

1. there exists a total function f such that f converts a consistent state of S_1 into a consistent state of S_2.
2. there exists a functions f' such that f' converts a consistent state of S_2 into a consistent states of S_1.
3. the composition of f followed by f' is the identity on the set of all consistent states of S_1.

If f' is also total and the composition of f' followed by f is the identity on the set of all consistent states of S_2, then S_1 and S_2 have *equivalent information capacity*.

Information capacity has been used extensively to characterise the correctness of schema conversions within the same data model or between data models. The degree of information capacity preservation depends on the following goal of schema conversion:

1. If schema S_2 is used only as a view for querying a specified database using schema *S1* then the conversion of S_1 into S_2 does not need to preserve the information capacity of S_1; however, if S_1 is also used for querying a database specified using schema S_2 then S_1 and S_2 must have equivalent information capacity;
2. If schema S_2 is used for viewing an entire specified database using S_1 then S_2 must have at least the information capacity of S_1;
3. If schema S_2 is used for updating a specified database using schema S_1 then S_1 and S_2 must have equivalent information capacity.

2.3.1 Drawbacks of schema conversion

Although the theoretical requirements for schema conversion mentioned in Section 2.2 meant to preserve the *information capacity* but in reality the outcome is not so uniform. For example Extended Entity Relations (EER) tools include translator which converts to relational schemas. These tools also support reverse conversions from relational schemas to EER. But when unnormalised relations are used for representing classes and objects, the schema translator fails to preserve equivalent

information capacity with the object schemas they represent. Sometimes additional constraints are required to ensure information capacity equivalent.

Schema converters that do not preserve information capacity can lead to distorted results. However, the information capacity equivalence criterion can sometimes be too strict so it loses the equivalent information capacity. For example, the reverse engineering or retrofitting object schemas in addition with the existing relational schemas sometimes add information which are missing in the converted relational schema, or ignore information that is not relevant for constructing the object schemas. This suggests that schema conversion needs to consider preserving *information capacity* regarding subsets of the schemas involved in the conversion. So, it requires to develop a criteria for determining the subsets that are relevant to schema conversion. Furthermore, reliable schema converters cannot be developed without knowing the precise syntactic and semantic definitions of the data models targeted for conversion. Such definitions are not always available.

2.4 Identifying the Present Problems in Interoperation

Molecular Biology DBs have been created by such a diverse set of international groups that nobody could really legislate the standards at any single level of abstraction, and even less at the existing levels of heterogeneity, such as the conceptualisation, the data model or the query language. Although the federated approach is the traditional approach within the computer science community, it has received little attention in the bioinformatics community (Karp, 1995).

Early attempts to manage heterogeneous databases were based on resolving heterogeneity by consolidating these databases either physically by integrating into a single homogenous database, or virtually by imposing a common data definition language, data model or even DBMS for heterogeneous databases. These attempts failed because they require a very difficult degree of attainment for co-operation and it also means a costly replacement of application that are already based on the existing database structures. The most effective way of coping with heterogeneous databases is to allow them to preserve their autonomy, *i.e.*, their local definitions, applications and policy of exchanging data with other databases. Markowitz (1995ab), suggested to connect the databases using hyper links or to organise them into database federations or multidatabase systems and to construct data warehouses for interoperability within the databases. Schonbach (2000) described the need of data warehousing to assist the analytical tasks in bioinformatics application. He emphasised on subject oriented data

warehouse in contrast to a general purpose database which will be comprised of facts (analysis of particular item) and which will be using same type of view. But combining databases using hyper links or creating data warehouses or database federations will have the following limitations:

I. If the database is linked using only hyper link system, then such system is limited to select a starting data item within one database and then follow the hyper links between data items within or across other databases. Multiple databases can be accessed by clicking on key words for which links have been provided. But the next set of data has relations with the preceding set of data and this approach does not allow any comparison. These types of transactions are independent of preceding transactions and thus they are stateless and discrete.

II. Database federation and data warehouse entail developing a global schema of the component heterogeneous database. Discrepancies between these definitions are resolved before they are integrated into one global schema. In data warehousing, data from component databases are loaded into central databases, therefore, it is creating data replication and data redundancy. Query processing is local to the warehouse and therefore it is restricted only to the data items that it contains. Moreover, if any of the component database is updated which is a subset of the data warehouse, data warehouse itself needs to be updated as a whole. Otherwise, it will lose data integrity. However, in a database federation, query translator converts queries expressed over the global schema to queries for component databases. In this case, biologists are required to express the queries directly using SQL, which they are not only reluctant to do but it is also hard to imagine that they would be willing to express queries in more powerful languages such as CPL. Therefore, it demands dynamic navigation within the component databases for data exploration and integrating the query result in a single interface.

III. Multidatabase systems are collections of loosely coupled databases that are not integrated using global schema. Querying a multidatabase system involves constructing queries over component databases, where a query explicitly refers to the elements of such databases participating

as component. Alternatively, a heterogeneous database system can be provided with a multidatabase query language that allows expressing queries which refer directly to elements of component databases. Biological researchers find these multidatabase query languages to be very complex to use, moreover, it requires prior knowledge of the data elements that resides in a particular component database.

Markowitz and Ritter (1995) and Markowitz and Topaloglou (2001) summarised that manipulating (*i.e.* updating) and maintaining (*i.e.* reorganising) a large database are inherently more complex process than for smaller component database. They also suggested some criteria for characterising a heterogeneous database system. Three of these criteria are of particular concern for integrating the biological databases. These are:

i. Database heterogeneity and cooperation assumptions
ii. Type of query interface and processing
iii. Extent of synchronisation with component database.

Letovsky (1995), identified several criteria for evaluating the scaling properties of an information system and then evaluate the options for the main technology which are available today according to these criteria. These parameters are:

i. *Performance parameter*: technological scalability and content scalability
ii. *Submission and Access scalability*
iii. *Recall*: fraction of relevant answers in a system that is retrieved by query
iv. *Precision*: fraction of retrieved answers that are relevant
v. *Schema partitioning*: each component database controls part of the schema
vi. *Data partitioning:* multiple component database contains data for some portions of the schema.

Letovsky (1995) summarised that these parameters can be achieved by increasing the decentralisation away from central databases and by developing subsets of the scientific community that has an interest in that particular information. At present, existing systems are compensating the lack of interoperation among the databases by allowing more complex queries in one facility, which increases the *precision* but at the expense of *recall.*

Robbins (1996) identified that creating collections of data resources perceived by users are the more challenging part. These databases will be integrated with each

other and it will maintain its autonomy at the same time, especially for basic creation and maintenance of its data resources. Davies *et al.* (1997) also described the need of subject oriented, small scale and dedicated databases. They implemented their system and integrated with other databases of that particular subject domain using hyper links. The major drawbacks of linking databases using hyper link have been already discussed, moreover, database interoperability with just one research domain will not be enough. It is essential that the users should be able to retrieve related data from multiple databases such as GDB, PDB and GenBank without submitting separate queries to the databases and the users should be able to integrate the results for more efficient and effective analysis of the data.

2.5 Research Issues

On the basis of the above review it is concluded that the following questions need to be addressed:

- How can the molecular biology databases be integrated while keeping all the technological aspects transparent to the users?
- How can the local autonomy of databases be preserved for data submission which will be maintained and created by local authority?
- How can queries be submitted to the resource databases without writing any complex script languages and how data can be explored dynamically using automated navigation across the resource databases?

2.5.1 Current research projects and their limitations

Karp and Suzane (1996) and Karp *et al.,* (1998) developed *Ecocyc* system which consists of a knowledge base (kb) that describes the genes and intermediary mechanism of *E.Coli* and a graphical user interface (GUI) for accessing this information. They described the design and the implementation of visual presentations. It offers hypertext navigation among related entities and multiple views of the same entity. They have integrated genomic data with comprehensive knowledge of the metabolic functions of gene products. They have used a fixed number of commonly used queries to explore the data of these two databases. Query facilities explore the databases using hypertext navigation. Hypertext navigation does not allow comprehensive comparison and thus it can not be used for data analysis. It does not also provide any knowledge or details of the data formats used in the database.

Davidson *et al.* (1997) have integrated gene expression patterns along with the phenotypes of mutant embryos. They have attempted to associate gene expression with forming structures, or with particular cellular activities. They have described the need of a database which will integrate information from a wide range of sources, including assays on dissected and homogenized parts of the embryo (*e.g.* RT-PCR (see Glossary), Northern and Western – blots (see Glossary), RNase protection assays, *etc.*). They have also emphasised that it would be able to store expression data at any resolution ranging from incomplete accounts at low resolution to complete descriptions, and it would allow information to be shared across the biomedical community. They have implemented spatial description of a series of digital 3D model of embryos and integrated with the Mouse Atlas database for anatomical correlation using hyperlink text. But they have not described how this technique could be used for database correlation with gene mapping and with other resource databases using image object keying (see Section 3.4). They have only allowed data exploration using a fixed text data clicking and it has not provided a wide range of query across the resource database for more meaningful analysis.

Medigue, *et al.* (1999) developed a system *Imagene* which provides a user interface to display the result produced by several tasks on the same screen. The system is actually based on genome sequence and annotation. Although they have claimed that this system provides a cooperative system for sequence analysis the system is mainly menu driven. User must have prior knowledge of the location of data sources. Navigation to multiple data sources can be conducted by using this location information to explore the data. This involves creating knowledge based wrapper and mediators and hence a very techno-centric approach. The user also needs to be familiar with the multiple 'manager' to combine the results and to navigate through the databases. It has used specific class hierarchy and biological entities to accept input. These class hierarchies are fixed and can not vary for carrying out different types of analysis.

Kemp *et al.* (2000b and 1996) used functional data model to integrate the biological data sources. They have used multi-database approach and a mediator to access other data sources. From their research, it is evident that mapping flat files onto functional data model entity classes and attributes is required in order to integrate data from existing databanks.

Siepel *et al.* (2001) developed a decentralised, component-based approach to integrate the heterogeneous bioinformatics sources, called ISYS. ISYS system is based on two basic components: *ClientBus* and *ISYS Client Environment (ICE)*. These independent components synchronise with other components of user interfaces to present the result. It is a set-building tools which are resulted as combinations of components. These tools are heavily dependent on global schema. The classes have been designed in abstract form to retain the compatibility with the other databases. Thus query is restricted and it is menu driven. Similarity search is based on object similarity within other databases. The data models used in ISYS system is less comprehensive and abstract in nature, hence synchronising with other databases depends heavily on the 'view' and core attributes of the component's data models.

The issues raised in the previous section and in this section have shown that the current research activities lack to provide a full range of solutions for data submission and to maintain its autonomy. It also fails to provide a complete framework for data correlation to act independently and dynamically which will navigate along multiple databases for automated data exploration without any prior knowledge of the location of the databases.

2.6 Summary

This chapter has started by looking at the heterogeneity nature of the molecular biology databases (Section 2.1). 'Database interoperability' has become a major issue in bioinformatics community for effective analysis of biological data. The strategies which have received most attention for integration are 'consolidation' and 'federation' approaches. 'Consolidation' approach refers to physical or even administrative consolidation of certain key databases. But it offers limited solution to database interoperability problem within molecular biology. 'Federation' approach is based on database linking using reference links or developing multidatabase system. It could also be data warehousing. Data warehouses entail developing a global schema (view) of the component MBDs, where definitions of these MBDs are expressed in a common DDL and discrepancies between these definitions are resolved first before they could be integrated into the global schema. But, generic classes or attributes are used for designing global view of the related databases which might not be able to capture full semantics of the data. Moreover, these data models are based on a

particular system domain, therefore, unable to capture all semantics of component data models.

Section 2.1 has also looked at issues like semantic conflicts. Resolving semantic conflicts can be achieved by renaming, rearranging or removing structural dissimilarities. Alternatively, users can resolve the semantic conflicts and can keep the record of resolution. But these approaches are not feasible due to the huge growth of the databases and due to legislative restriction to access the data structure of the major data sources.

The interfaces that have been provided for data access do not provide any information regarding the data structure of the database. It does not also provide any interpretation of the query results. One specific example is *Entrez*, which provides multi-range queries but it lacks data interpretation. Moreover, the query is specific to the particular attributes of the database. *Entrez* does not provide any automated navigation for accessing multiple database to explore relevant data and to compare the results for further analysis.

Although, *Entrez* is a widely used interface to integrate with major databases for data exploration, it has failed to develop a cooperative environment for data correlation.

Another popular approach of database integration is to convert multiple schema into one global schema (Section 2.1.4). This approach tends to distort result or fails to preserve *equivalent information capacity*. It also develops derived construct for the schema which leads to the overlapping of attributes and to information redundancy.

Section 2.2 has looked at integration issues. The section has reviewed the basic structure of the molecular biology database schemas and it has highlighted that there is no definite indication of which type of data model is preferable for storing molecular biology data.

The section has also examined the criteria and parameters for molecular biology database integration. It has established that these criteria can be achieved by developing decentralized and bottom-up approach for database integration. It has also confirmed that higher degree of *precision* and *recall* can be achieved by developing cooperation of databases in a particular framework. Section 2.3 has then looked at the problems faced in interoperation.

The review and the critical analysis carried out in these sections has resulted in forming a set of research issues and questions (Section 2.5) which need to be addressed for effective and meaningful analysis of biological data. A number of existing research projects (Section 2.4.1) attempted to overcome these issues but most of these projects have not achieved any high degree of satisfaction for 'database interoperability'.

Difficulties of data integration can not be resolved through data consolidation. The data integration issue can be addressed by creating distinct, officially sanctioned subsets of data resources relevant to individual research areas. This is because it will not be possible to identify a set of information resources which will be relevant to the whole research community. Most of the genome database projects have focused on the community domain and they have not specifically addressed the local-user domain. Developing specific and non-generic systems are usually more cost-effective and may lead to more rapid support of local end-users. Integration of relevant data using object keying will provide an aid to the biological researcher to combine the result and to analyse the data for more meaningful information. Despite a lot of effort made by a number of researchers, a cooperative framework is yet to be developed to provide more synchronisation between the component and heterogeneous databases.

The next chapter will discuss the proposed approach to address the research issues that have been raised in this chapter. The research proposes a component database model and suggests a multi-agent based cooperative environment for database integration.

Chapter 3

Conceptual Modeling of Gene Mutation Data

Chapter Objective

At the beginning of this chapter the argument for using task specific and dedicated component database for database integration is established. A task oriented database for gene mutation data is selected as a component database. The existing Gene Mutation Database represents a rich source of information, however, it lacks the complete data sets which is essential to analyse the trait for indirect association of diseases. To understand the extent of polymorphism present in human gene requires examining the inheritance patterns. The parameters that need to be considered to analyse the disease are the ethnic origin, frequency of abnormalities, genetic distance between loci (d), mutation rate (μ) and correlation with reported clinical importance. On the basis of these analysis the conceptual schemas for the following mutation data have been developed: trait analysis, gene mutation data, laboratory data and pathological lesions data. The chapter then derives a non-redundant schema integration approach for interoperating with other molecular biology databases for more meaningful information.

Chapter Contents

Chapter 3

Conceptual Modelling of Gene Mutation Data

If the (bioinformatics) work can be split into several independent tasks, these can either be processed by a single powerful system one by one, two or more at a time if several processors are available, or they can be distributed among several systems (distributed computing)

Jain (2002), Distributed Computing in Bioinformatics, Applied Bioinformatics :1(1)

3.1 Introduction

The types of problems and difficulties which exist for integration of heterogeneous databases have been looked in the previous chapter. This chapter addresses an extensible framework for capturing laboratory data with its meta-data properties in a semistructured data model. This chapter also focuses on the issues of creating a local and task specific component database which is a subset of global data resources.

At present no single, unambiguous, complete and static model (Geihs, 2001) exists for storing gene mutation data for analysing indirect association of diseases. This chapter presents a genetic disorder database schema for variance analysis. It also proposes a framework to allow the component database to act independently in order to take part in integration process without affecting the local autonomy. The other relevant issues for creating molecular biology database are:

i. data submission
ii. how to avoid attribute redundancy for database integration
iii. to provide a web based framework for multidatabase query and interoperation.

The Section 3.2 explores the present approaches used for data submission into data banks. The section argues that the present approach can be replaced by introducing component databases for each laboratory which is the primary source of the data. A task specific and dedicated database is suggested for storing local data. The role of the task specific database and their management are described in section 3.2.1 and section 3.2.2.

The research is taken the case of genetic disorder as a domain for the implementation part. The reasons for choosing this domain and a set of details of the schema of this domain are presented in section 3.3. Section 3.4 presents the implementation environment of the schemas and it also describes the framework for interoperation with other component databases. Section 3.5 summarises the issues presented in this chapter.

3.2 Data Submission to the Genome Databases

Initially all the databases stored data on the basis of reviewing papers. An expert will read the literature to extract the required information and it will also interpret the previous published results. But as the volume of data increased, new methods emerged to accelerate the process. One method to speed up journal scanning is the use of automated scanning procedures. Another method is to allow researchers to submit their data directly to the databases. In the second case it requires editorial review and other quality controls, *i.e.*, searching for same data, over sequence data *etc.* Results of these analysis could be returned to the researcher who could then respond with an improved version for submission as appropriate. This effectively replaces the three error-generating steps: preparing the data for sending to the journal, typesetting the journal, and scanning the journal with one error-correcting step (Krawczak *et al.* 2000). Direct data submission in comparison with journal scanning, gives better and faster data transfer results. However, the problem with this method of submission is that these data need to be published and the researchers are unable to submit the full set of results and photomicrographs because the interfaces used for data capturing have limitations. For example, HGMD databases does not deal with image data formats. Moreover, in many genome research, data are needed to be submitted to multiple databases. A coordinated research program of a particular laboratory publishes data through a number of different databases such as GenBank, PDB, GDB and others. These laboratories would be unnecessarily burdened if they require to prepare their data for multiple databases and to take responsibility for checking the validity of links among the related elements in all the databases. Developing and perfecting a coordinated direct data-submission method for all genome databases must be given a high priority. To resolve these issues and to submit data in detail, the research is proposing to create a component database which

will be a component of the whole federated database. This database will be small-scale, laboratory based and dedicated which will store all the laboratory results and it will be coordinating with other community databases. This will enable the biological researchers to be able to submit the results with a single electronic transaction which can then be coordinated automatically with multiple relevant data sources. This method of data submission can be applied to a broad spectrum of the research community starting from low-volume to high-volume data producers.

3.2.1 Component database for interoperability

One of the novel features of a small scale, laboratory-based, dedicated and open-ended genetic disorder database is the ability to investigate the gene function by integrating the independent descriptions of the protein morphology structure as well as their changes in the protein structure. The protein structure changes due to the abnormalities in gene expression. A data model needs to be developed without overlapping the objects or relations of component global schema. Data searching will start from laboratory based local component database and it will continue searching the relevant information through one or more global databases. A meta data will be provided for schema translation and data conversion as a component of data model. The meta data will provide the data model and the data structure information. This meta data will then be able to extend the synchronisation among the component and global databases for effective interoperability. It will assist the biological researchers to interpret the query within its own local database which will automatically link the global database for other relevant information associated with it (Khan *et al.*, 2001a).

3.2.2 Component database maintenance

Query processing in data warehousing is complex in nature because of the existence of complex and huge volume of data. Biologist with average computer skills find it difficult to maintain such a complex database. A small database is much easier to maintain and to manipulate without affecting the global transactions. The content of a small scale and specialised database can also be much more detailed than any other global database. The other aspect of a small database is its close relationship with the data warehouse concept. Data warehousing requires to develop a global schema of the component molecular biology databases (MBD). To create these global schemas the discrepancies between the component MBD needs to be

resolved first they are expressed in a common data definition. The small-scale and subject oriented database deals with a detailed description of the subject which consist of attributes related to that particular domain. Therefore, a small-scale database can be used as component database for local query interpretation and it can then be linked with other data warehouses for global query, thus giving it a generic nature. The main objective here is that the biologists with minimum computer literacy should be able to maintain and manipulate such a database (Khan *et al.*, 2001).

3.3. Genetic Disorder Database as Component Database

The research aims to store the genetic disorder data in a database which will act as component database of a federated system. The following sections explain how this component database can be used to initiate coordination among the related databases. Reasons for choosing gene mutation database as component database have also been explained here.

Human Gene Mutation Database (HGMD) was developed to store gene mutation information which holds more than 12,900 different lesions (Krawczak *et al.* 2000). But this database lacks complete data sets which are essential for understanding the underlying molecular mechanism of diseases in biomedical and drug development related research. An example in this context would be mutation data classification and the hyper mutable sites identification (Cooper and Krawczak, 1993 and Cooper *et al.*, 1999) which are not included in HGMD database. This information, *i.e.* mutation data classification, hyper mutable sites *etc.*, are based on published data collected from different journals. The implementation process was carried out by a manual and computerised search which can scan 250 journals on a weekly/monthly basis (Krawczak *et al.*, 2000). But, HGMD entries are still limited to original published reports. Although some data are taken from 'Mutation updates' or review articles, mutations which are reported in abstract form and which are not necessarily disease causing but instead associated with disease phenotypes and variants have not been included in HGMD. Moreover, data extraction from journals requires standard formulation for information representation which might avoid various exceptions that exist in genetic mutation studies.

Identifying and understanding the properties of genome, genome mutation

and conducting comparison of genomes and its products depend upon the provision of storing, sharing and analysing the complete genomic data sets. For example, mutation rate, mutation frequency, or protein images are fundamental data sets for comparing with wild type proteins and for variance analysis of diseases. This research proposes a complete schema (Section 3.3.3) for storing the complete set of gene mutation data which will be used for genetic variance analysis (Khan and Rahman, 2002a).

3.3.1 Parameters used in schema design

Examining the inheritance patterns in human can determine the extend of polymorphism. In fact these inheritance patterns in human identify the potential recombinants which are present in human gene *loci*. Evidence in favour of or against a recombination fraction, say θ, is expressed by the quantity $Z(\theta)$, termed as *lod score* (Cooper and Krawczak, 1993). This is defined by the following expression:

$$Z(\theta_1) = \log_{10}[L\ (\text{data}:\theta_1)/L\ (\text{data}:\ \theta = 0.5)]$$

Here $L(data:\ \theta_1)$ denotes the likelihood of the observed genotype or phenotype data, $L(data:\ \theta = 0.5)$ is the likelihood assuming that θ equals *0.5, i.e.*, two *loci* are unlinked.

However, sex differences will have an impact in determining the likelihood of disease (θ) in human. Therefore, performing two distinct analysis on recombination fraction based on two distinct parameters θ_m (for male) and θ_f (for female) will produce different recombination fraction for male and for female (Rao *et al.*, 1978).

Indirect association of diseases in human can be analysed by measuring the quantity at which a particular dinucleotide is affected by point mutation at one of its two bases. This parameter is called relative dinucleotide mutabilities (*rdm*). If δ and δ' denote two *dinucleotides* which differ exactly by one position, then the *rdm* can be defined as:

$$\text{rdm}(\delta \rightarrow \delta') = O\ (\delta \rightarrow \delta')\ /\ E(\delta \rightarrow \delta')$$

where O is the observed frequency of $\delta \rightarrow \delta'$ among the point mutations. E is a real number proportional to the expected frequency, assuming that all dinucleotide mutations are equally likely. Values of E depends on the probability of clinical detection.

Inheritance pattern in human can also be determined by mutation rate (μ).

This mutation rate is conventionally defined as the ratio of the number of germ cells carrying a particular *de novo* mutation to the total number of germ cells being at risk for carrying that mutation (Vogel, 1990).

3.3.2 Designing gene mutation data models

In order to analyse the genetic variance of human, all the parameters described in Section 3.3.1 need to be stored so that indirect association of disease in human could be described. The emphasis here is on presenting indirect association of disease data rather than presenting the direct association of disease information which can be accessed from HGMD. The data models for depicting indirect association of diseases based on Unified Modelling Language (Booch *et al.,* 1999) are presented here. Genetic trait analysis data, mutation type data, laboratory data for mutation study and pathological lesions data have been modelled using Unified Modelling Language (UML), where the classes have been drawn as rectangles with the name of classes within the rectangles. *Generalizations* have been represented by a line with an arrowhead (Discriminator). Relationships between classes have been represented by lines, with the name of the role that the class plays in the relationship written adjacent to the line, along with the multiplicity, which indicates the number of objects that may participate in the relationship. When the classes depict an a*ggregation* then a line with a diamond at the end has been used. The conceptual level of ANSI-SPARC (see Glossary) structure of database is an independent level. The implementation of conceptual level depends neither on software nor on the platform of implementation. External and Internal schemas are developed on the basis of conceptual level. The conceptual diagram of the gene mutation data will provide facilities for wide range of analysis (Khan and Rahman 2002a).

3.3.2.1 Genetic trait analysis model

A schema for storing the data for inheritance pattern analysis is proposed in Figure 3.1. The following information are required to be stored for inheritance pattern analysis: recombination fraction, frequency of abnormalities and relative dinucleotide mutabilities. Trait analysis is the aggregation of gene expression and inheritance pattern results. Inheritance pattern has the following subtypes: background, ethnicity, geographical region, number of cases studied, types of genetic

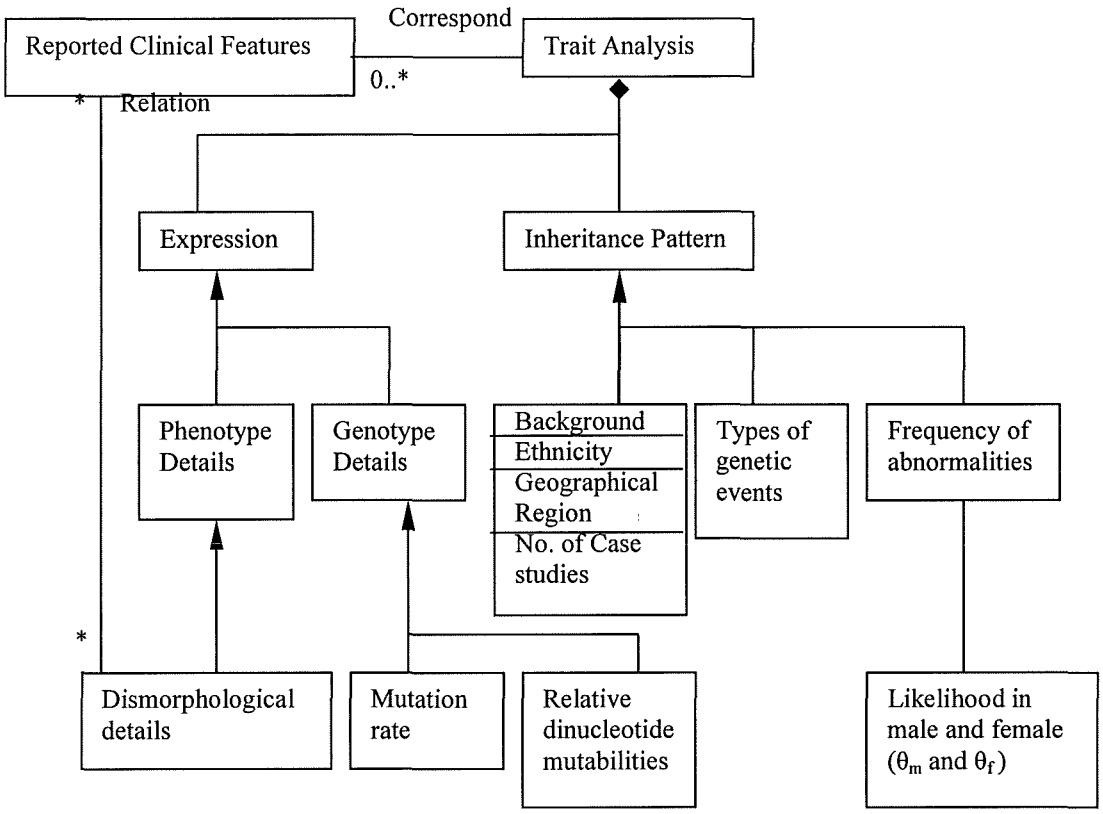


Figure 3.1 Inheritance pattern data model.

events and frequency of abnormalities. We need to store background, ethnicity and geographical region because genetic expression varies from one geographical region to another geographical region. For example, *dyslipidaemia* gene falls in the same *loci* to which diabetes and obesity genes have been mapped for French, Finnish and North American populations, but the same *loci* might not be the root for this disorder in population from other geographical locations. Trait analysis might have none or multiple corresponding reported clinical features. Many *dismorphological* details can be reported as clinical features by different groups of researchers. The reported *dismorphological* details can then be correlated with gene mutation data of the component database (Khan and Rahman, 2002a).

3.3.2.2 Classification of gene mutation data model

This section describes the object model which will categorise the gene mutation data. There are two types of mutations: mutation due to the production of less synthesised gene products (polypeptides) and mutations due to the production of abnormal gene products. Figure 3.2 proposes the conceptual schema for classification

of gene mutation data. Amount of synthesis of gene products can be affected if transcription or translation is affected. The other reason which might affect the amount of synthesis of gene products is malfunctioning of gene structure or promoter function. Transcription region of mRNA can be affected due to the mutation at transcriptional site, splice junction, *3′ UTR* (3′ untranslated region) mutation or due to the *polyadenylation* at transcription site. On the other hand translation can be affected when there is any mutation at initiation codon (see Glossary), termination codon (see Glossary), nonsense mutation (see Glossary) or at *5′ UTR*. Abnormal gene product can also be produced when two genes are fused together, or if there is any defective post translational processing. Abnormal gene product can also be produced if gene products are shortened or gene products are elongated. Shortened gene product and elongated gene product might be caused due to deletion or frame shift in a particular gene. Deletion, frameshift and insertion can also affect the gene structure. Insertion has several subtypes and these are small insertion, duplication, inversion unstable repeat sequence and large insertion. Small insertion changes the gene by adding 1-5 new neucleotides within the gene sequence. It can be caused by slipped mispairing or can be mediated by inverted repeat or symmetric elements (Khan and Rahman, 2002a).

3.3.2.3 Laboratory data model

It is necessary to do experiment on restriction fragment length polymorphism (RFLP) to analyse the indirect association of diseases. In this experiment, if we compare the restriction maps of the DNA sequences of the relevant alleles (see Glossary), they can be polymorphic in the sense that each map or sequence will be different from the others. Although not evident from the phenotype (see Glossary), the wild type may itself be polymorphic. Multiple versions of the wild-type allele may be distinguished by differences in sequence that do not affect their function, and which therefore are not detected in the form of phenotypic variants. Some polymorphisms in the genome can be detected by comparing the restriction maps of different individuals. The criterion is a change in the pattern of fragments produced by cleavage with a restriction enzyme. As the restriction map (see Glossary) is independent of gene function, a polymorphism at this level can be detected irrespective of whether the sequence change affects the phenotype. This RFLP

technique (see Glossary) is used as genetic marker, by which genotype variance can be examined instead of assessing any phenotypic variation. Figure 3.3 shows the proposed data model for laboratory evidence. In RFLP experiment each gel electrophoresis is compared with corresponding references. For each RFLP a restriction enzyme (see Glossary) is used in a controlled environment. In gel electrophoresis, labelling is quantified by measuring intensity of blotting or staining (Khan and Rahman 2002a).

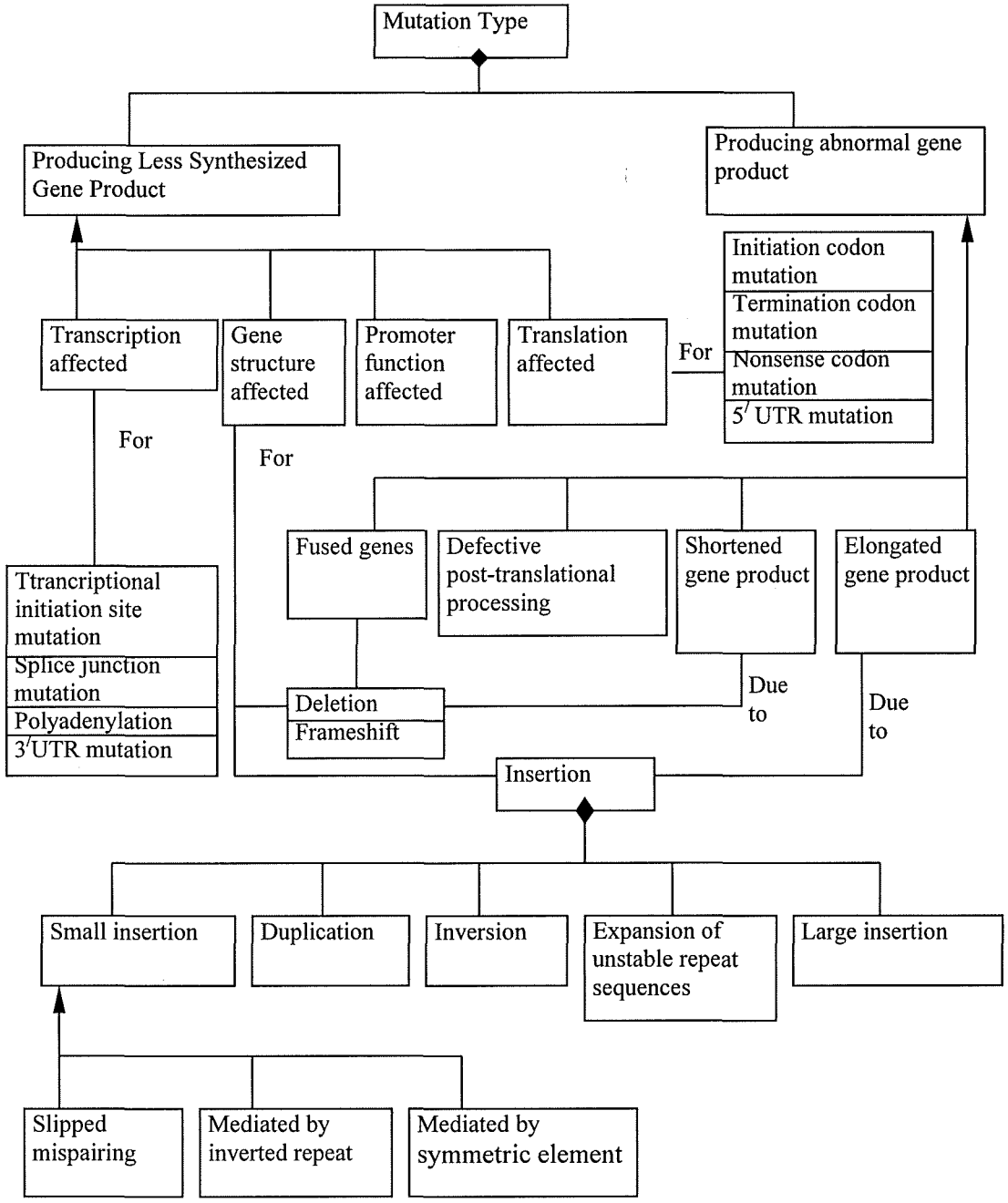


Figure 3.2 Gene mutation data model.

3.3.2.4 Pathological lesions data model

Pathological lesions might be caused due to deletion, insertion or inversion. Base pair substitution and abnormalities in specific repeat sequences could also lead to the pathological lesions. Base pair substitution has several subtypes: single base pair, or multiple base pair. A class diagram for pathological lesions data modelling is proposed in Figure 3.4. It is required to store the pathological lesions data for future clinical references. A single base pair substitution is referred to any modification in gene due to substitution by a single nucleotide. This can take place at coding region, at splice sites, or at promoter region. Multiple base pair substitutions classes will store the information which ultimately will lead to the gross change of gene expression (Khan and Rahman 2002a).

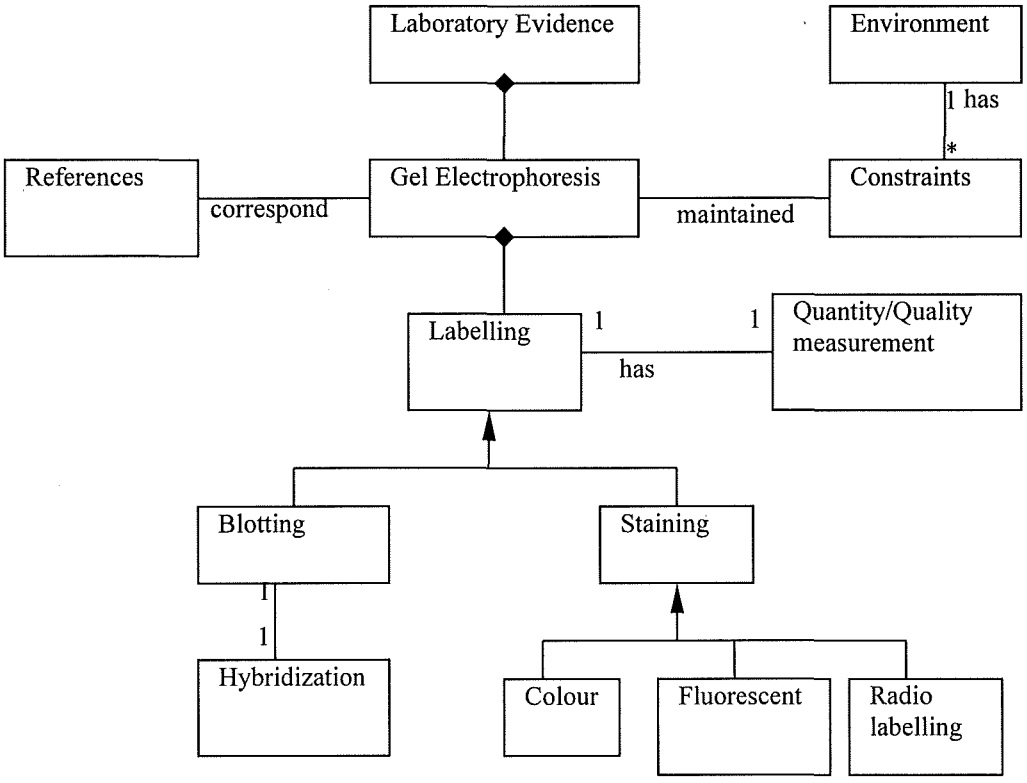


Figure 3.3 Empirical data model.

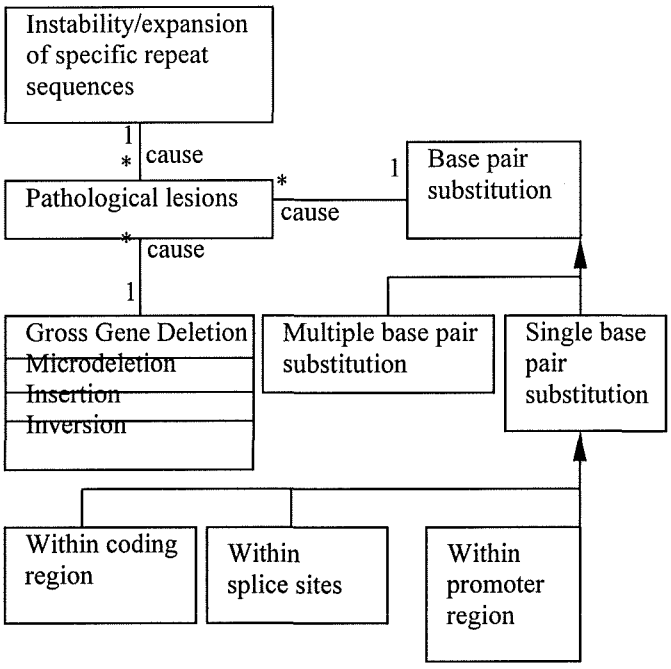


Figure 3.4 Pathological lesions data model

3.4 Implementation of the Schema

The implementation of dedicated and task specific database in the context of a particular laboratory needs to focus on the following factors:

- appropriate data model to store all the relevant data of the laboratory: it should ensure that the data definition language accepts the descriptive definition instead of the procedural constructs only.
- ability to communicate with other data models for data exchange and interoperability: it should ensure that the data can be exchanged through HTTP protocols and can also use other form of database structure, *i.e.* relational or object oriented, for query formulation.
- appropriate data structure and inheritance for efficient query management: it should ensure the appropriate domain values coping with molecular biology data, complex and user defined domain values and multiple inheritance for object abstraction.
- adding appropriate semantics to the data model to resolve semantic conflicts: it should ensure the proper annotation of the elements and resources for dealing with semantic conflicts.

- appropriate database management so that the database can act as component of federated data resources: it should ensure that the data transactions, mediators and wrappers can be created effectively to include the database as a component of federated information system.
- appropriate view management for integrating component database with the federated database system: it should ensure that a global view can be created to reflect the participating resources schema for integration purpose.

Implementing the schema in relational or object relational model has drawbacks when it is to be used in the domain of medical data. Sheu *et al.* (2000) highlighted the limitations of using relational model. They pointed out that SQL has limited scope to make query on data, for example, finding any particular data corresponding to any behaviour. Moreover, SQL requires the formal representation of data in table relational form which is not always possible for molecular biology data. Connolly and Begg (2002) also described further limitations of the relational data model. They pointed out that the relational model relies on the homogeneous data structure, but conceptualising molecular biology data which are distributed under different platforms need to accept heterogeneous data. So, problems can not be solved by homogenising the data structure as it will increase the problem in several fold.

A partial solution of avoiding relational data model is to use object relational model to represent the laboratory data but it has its own drawbacks. It is not possible to describe multiple inheritance and it lacks describing complex data types such as sets, lists, *etc.* which is suitable for molecular biology data.

The next section describes the use of appropriate environment to implement the schema which will overcome the above drawbacks.

3.4.1 Environment for implementing the schemas

Davidson *et al.*, (1995 and 1999), highlighted that the schemas within the domain of molecular biology evolve rapidly in response to changing requirements and experimental techniques. Therefore, creating laboratory databases is far more challenging than creating databases for business environment. The laboratory data is

irregular and in general they are incomplete. It also has the potential of rapid and unpredictable changes. Data with such characteristics are termed as 'Semistructured data' (Connolly, 2002).

The molecular biology data needs to be web dependent so that it can have an immediate access and to cope with disparate databases. This makes it almost impossible to fit in any proper schema model like relational or object oriented model. The schema in proper relational or object oriented model places constraints on the data to make it interoperable with each other (Connolly, 2002).

There are a number of database management system to describe and manage the semistructured data, *e.g.* LORE (Lightweight Object Repository, McHugh *et al.* 1997) and XML (XML, 2000). XML is the mostly used database management system for semistructured data which allows designers to create their own customised tags to provide functionality. XML allows decompositions of objects into atomic and complex description. The complex objects can also have children objects and a single object can have multiple parent objects. An arbitrary complex network can be constructed in XML to model the relationship among the data.

The schemas proposed in section 3.3 are implemented in XML and the following tables describe the implementation details of these XML schemas.

Schema: LaboratoryEvidence.xsd

Elements	**Complex type**
Blottings	GelelectrophoresisType
Constraints	LabellingType
Laboratory_Evidence	Laboratory_EvidenceType
Staining	string

Element : Blotting

namespace	http://my_laboratory_evidence.com/namespace
Used by	ComplexType LabellingType
Source	<xs:element name="Blooting">

Element: Constraints

Namespace	http://my_laboratory_evidence.com/namespace
Used by	ComplexType GelelectrophoresisType

Attributes	**Name**	**Type**
	Protocol	xs:string
	Environment	xs:string

Source

```
<xs:element name="Constraints">
<xs:complexType>
  <xs:attribute name="Protocol" type="xs:string" />
  <xs:attribute name="Environment" type="xs:string"/>
 </xs:complexType>
</xs:element>
```

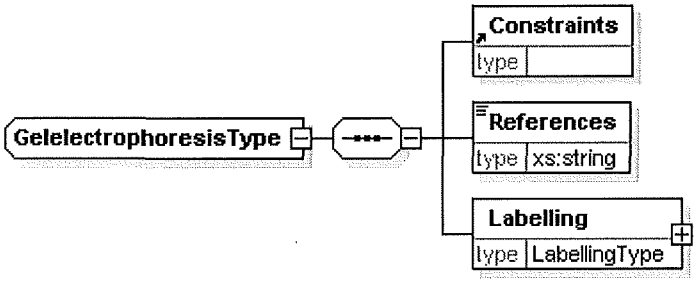
Element: Laboratory_Evidence

namespace http://my_laboratory_evidence.com/namespace
Used by LaboratoryEvidenceType
Children Gelelectrophoresis
Annotation store the laboratory evidence
Source

```
<xs:element name="Laboratory_Evidence"
type="LaboratoryEvidenceType">
<xs:annotation>
  <xs:documentation>store the laboratory
       evidence</xs:documentation>
 </xs:annotation>
</xs:element>
```

Element: Staining

namespace http://my_laboratory_evidence.com/namespace
Used by ComplexType LabellingType
Source <xs:element name="Staining"/>



CompelxType: GelelectrophoresisType

namespace http://my_laboratory_evidence.com/namespace
Used by Element LaboratoryEvidenceType/Gelelectrophoresis
Children Constraints References Labelling
Source

```
<xs:complexType name="GelelectrophoresisType">
  <xs:sequence>
  <xs:element ref="Constraints"/>
  <xs:element name="References" type="xs:string"/>
  <xs:element name="Labelling" type="LabellingType"/>
  </xs:sequence>
 </xs:complexType>
```

Element: Gelelectrophoresistype/References

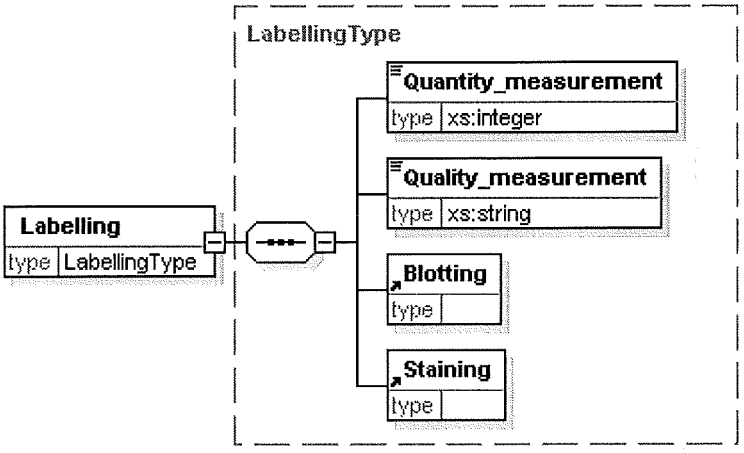
namespace http://my_laboratory_evidence.com/namespace
type xs:string

Source <xs:element name="References" type="xs:string"/>

Element: GelelectrophresisType/Labelling

namespace http://my_laboratory_evidence.com/namespace
type LabellingType
Children Quantity_measurement, Quality_measurement, Blotting, Staining
Source <xs:element name="Labelling" type="LabellingType"/>

ComplexType: LabellingType



namespace http://my_laboratory_evidence.com/namespace
Used by Element GelelectrophoresisType/Labelling
Children Quantity_measurement, Quality_measurement, Blotting, Staining
Source

```
<xs:complexType name="LabellingType">
  <xs:sequence>
  <xs:element name="Quantity_measurement" type="xs:integer"/>
  <xs:element name="Quality_measurement" type="xs:string"/>
  <xs:element ref="Blotting"/>
  <xs:element ref="Staining"/>
  </xs:sequence>
</xs:complexType>
```

Element: LabellingType/Quantity_measurement

namespace http://my_laboratory_evidence.com/namespace
type xs:string
Source <xs:element name="Quantity_measurement" type="xs:integer"/>

Element: LabellingType/Quality_measurement

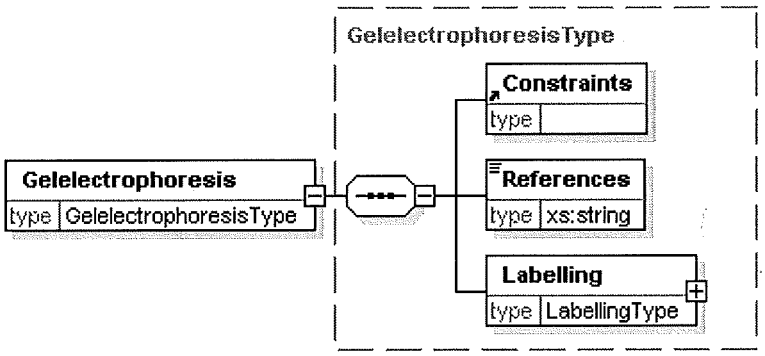
namespace http://my_laboratory_evidence.com/namespace
type xs:string
Source <xs:element name="Quality_measurement" type="xs:integer"/>

ComplexType: Laboratory_EvidenceType

namespace http://my_laboratory_evidence.com/namespace

Used by Element Laboratory_Evidence
Children Gelelectrophoresis
Source
```
<xs:complexType name="Laboratory_EvidenceType">
  <xs:sequence>
  <xs:element name="Gelelectrophoresis" type="GelelectrophoresisType"/>
  </xs:sequence>
</xs:complexType>
```

Element: Laboratory_evidencetype/Gelelectrophoresis

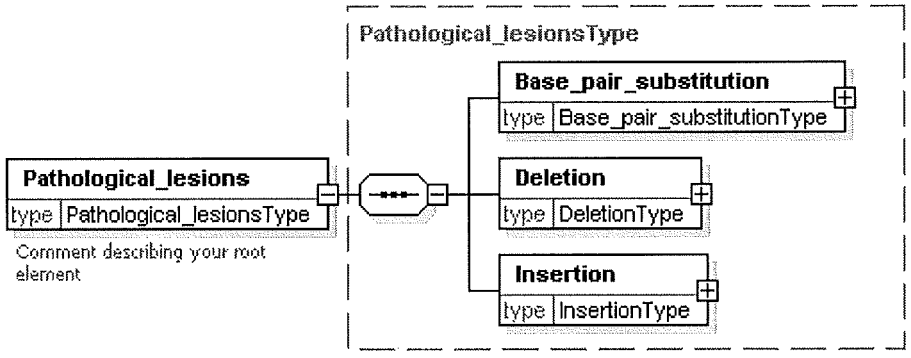


namespace http://my_laboratory_evidence.com/namespace
type GelelectrophoresisType
Children Constraints, References, Labelling
Source <xs:element name="Gelelectrophoresis" type="GelelectrophoresisType"/>

Schema PathologicalLessions.xsd

Elements	Complex type
Pathological_lesions	Base-pair_substitutionType DeletionType InsertionType Single_base_pair_substitutionType Small_base_pair_substitutionType Pathological_lessionType

element Pathological_lesions

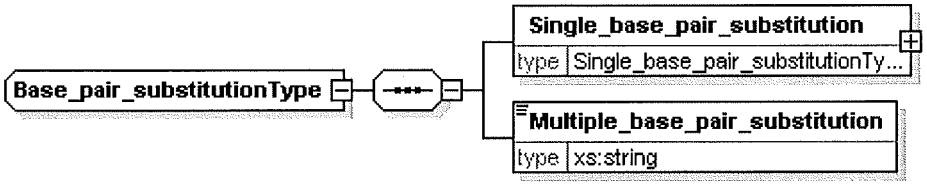


namespace http://my_pathological_lesions.com/namespace
type Pathological_lesionsType
children Base_pair_substitution Deletion Insertion
annotation documentation Comment describing your root element

source

```
<xs:element name="Pathological_lesions"
type="Pathological_lesionsType">
<xs:annotation>
   <xs:documentation>Comment describing your type of
   pathological_lesion</xs:documentation>
 </xs:annotation>
</xs:element>
```

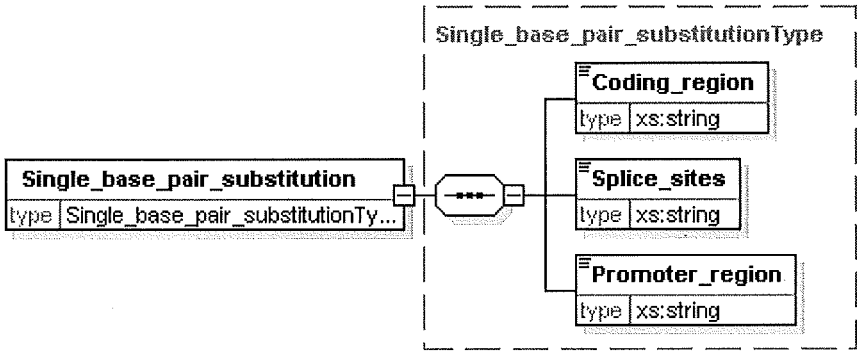
complexType Base_pair_substitutionType



namespace http://my_pathological_lesions.com/namespace
children Single_base_pair_substitution Multiple_base_pair_substitution
used by element Pathological_lesionsType/Base_pair_substitution
source

```
<xs:complexType name="Base_pair_substitutionType">
<xs:sequence>
        <xs:element name="Single_base_pair_substitution"
        type="Single_base_pair_substitutionType"/>
        <xs:element name="Multiple_base_pair_substitution"
        type="xs:string"/>
</xs:sequence>
</xs:complexType>
```

element Base_pair_substitutionType/Single_base_pair_substitution



namespace http://my_pathological_lesions.com/namespace
type Single_base_pair_substitutionType
children Coding_region Splice_sites Promoter_region
source

```
<xs:element name="Single_base_pair_substitution"
type="Single_base_pair_substitutionType"/>
```

element Base_pair_substitutionType/Multiple_base_pair_substitution
namespace http://my_pathological_lesions.com/namespace
type xs:string
source

```
<xs:element name="Multiple_base_pair_substitution"
type="xs:string"/>
```

complexType DeletionType
namespace http://my_pathological_lesions.com/namespace

children Gross_gene_deletion Microdeletion
used by element Pathological_lesionsType/Deletion
source

```
<xs:complexType name="DeletionType">
 <xs:sequence>
  <xs:element name="Gross_gene_deletion" type="xs:string"/>
  <xs:element name="Microdeletion" type="xs:string"/>
 </xs:sequence>
</xs:complexType>
```

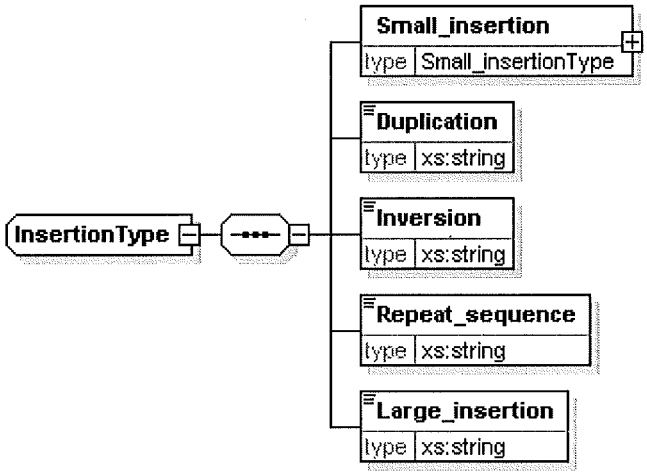
element DeletionType/Gross_gene_deletion diagram
namespace http://my_pathological_lesions.com/namespace
type xs:string
source

```
<xs:element name="Gross_gene_deletion" type="xs:string"/>
```

element DeletionType/Microdeletion diagram
namespace http://my_pathological_lesions.com/namespace
type xs:string
source

```
<xs:element name="Microdeletion" type="xs:string"/>
```

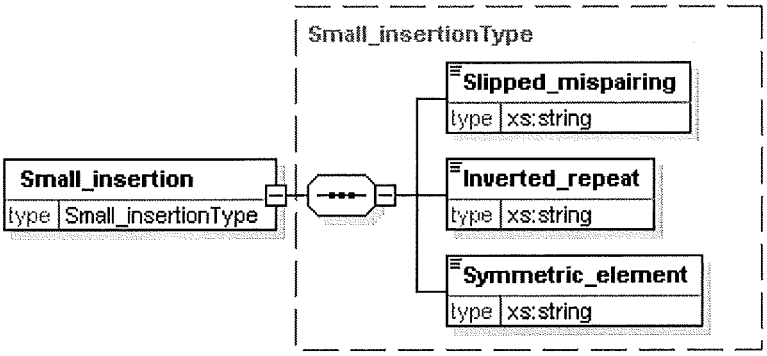
complexType InsertionType



namespace http://my_pathological_lesions.com/namespace
children Small_insertion Duplication Inversion Repeat_sequence Large_insertion
used by element Pathological_lesionsType/Insertion
source

```
<xs:complexType name="InsertionType">
 <xs:sequence>
        <xs:element name="Small_insertion"
        type="Small_insertionType"/>
  <xs:element name="Duplication" type="xs:string"/>
  <xs:element name="Inversion" type="xs:string"/>
  <xs:element name="Repeat_sequence" type="xs:string"/>
  <xs:element name="Large_insertion" type="xs:string"/>
 </xs:sequence>
</xs:complexType>
```

element InsertionType/Small_insertion



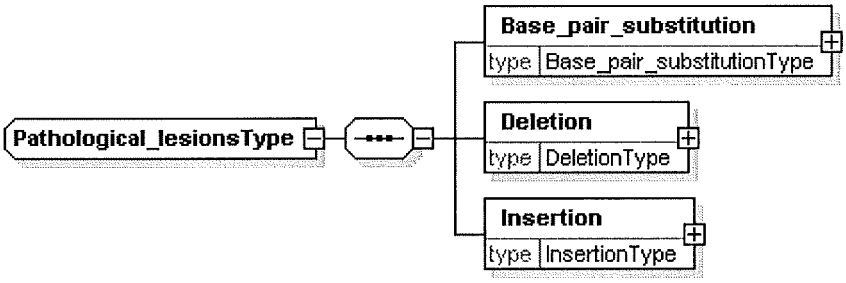
namespace http://my_pathological_lesions.com/namespace
type Small_insertionType
children Slipped_mispairing Inverted_repeat Symmetric_element
source <xs:element name="Small_insertion" type="Small_insertionType"/>

element InsertionType/Duplication diagram
namespace http://my_pathological_lesions.com/namespace
type xs:string
source <xs:element name="Duplication" type="xs:string"/>

element InsertionType/Inversion diagram
namespace http://my_pathological_lesions.com/namespace
type xs:string
source <xs:element name="Inversion" type="xs:string"/>

element InsertionType/Repeat_sequence
namespace http://my_pathological_lesions.com/namespace
type xs:string
source <xs:element name="Repeat_sequence" type="xs:string"/>

element InsertionType/Large_insertion diagram
namespace http://my_pathological_lesions.com/namespace
type xs:string
source <xs:element name="Large_insertion" type="xs:string"/>

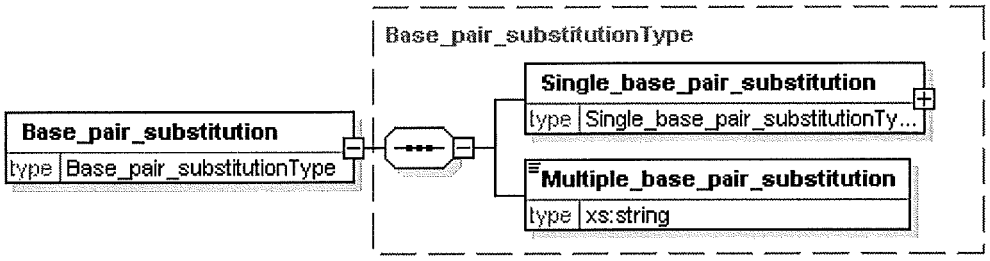


complexType Pathological_lesionsType

namespace http://my_pathological_lesions.com/namespace
children Base_pair_substitution Deletion Insertion
used by element Pathological_lesions
source <xs:complexType name="Pathological_lesionsType">
<xs:sequence>

```
    <xs:element name="Base_pair_substitution"
          type="Base_pair_substitutionType"/>
    <xs:element name="Deletion" type="DeletionType"/>
    <xs:element name="Insertion" type="InsertionType"/>
  </xs:sequence>
</xs:complexType>
```

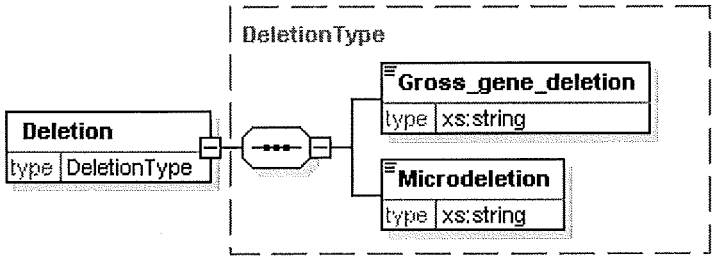
element Pathological_lesionsType/Base_pair_substitution



namespace http://my_pathological_lesions.com/namespace
type Base_pair_substitutionType
children Single_base_pair_substitution Multiple_base_pair_substitution
source

```
<xs:element name="Base_pair_substitution"
      type="Base_pair_substitutionType"/>
```

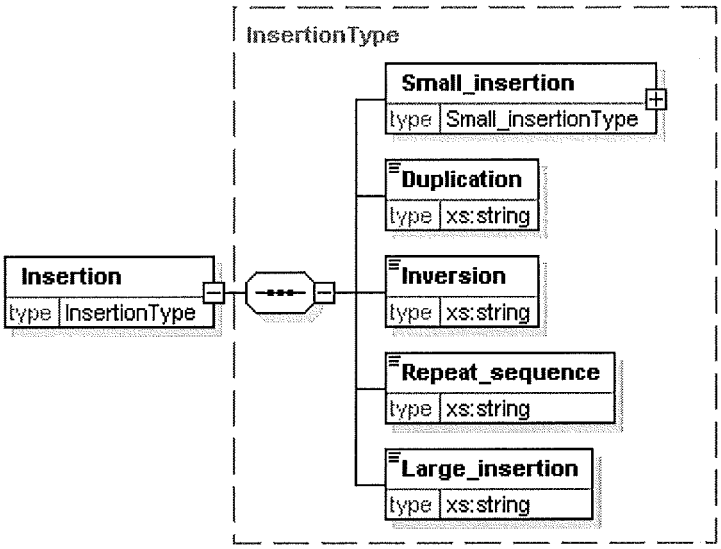
element Pathological_lesionsType/Deletion



namespace http://my_pathological_lesions.com/namespace
type DeletionType
children Gross_gene_deletion Microdeletion
source

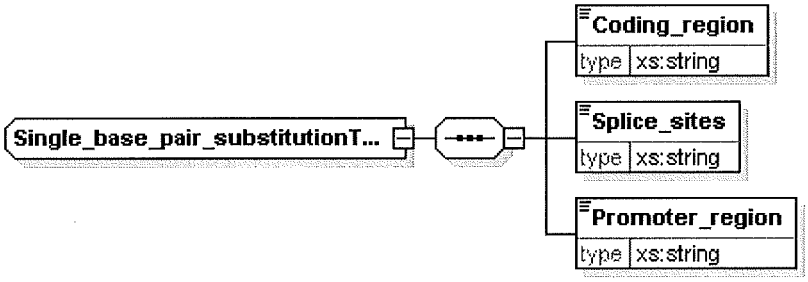
```
<xs:element name="Deletion" type="DeletionType"/>
```

element Pathological_lesionsType/Insertion



namespace http://my_pathological_lesions.com/namespace
type InsertionType
children Small_insertion Duplication Inversion Repeat_sequence Large_insertion
Source <xs:element name="Insertion" type="InsertionType"/>

complexType Single_base_pair_substitutionType



namespace http://my_pathological_lesions.com/namespace
children Coding_region Splice_sites Promoter_region
used by Base_pair_substitutionType/Single_base_pair_substitution
source

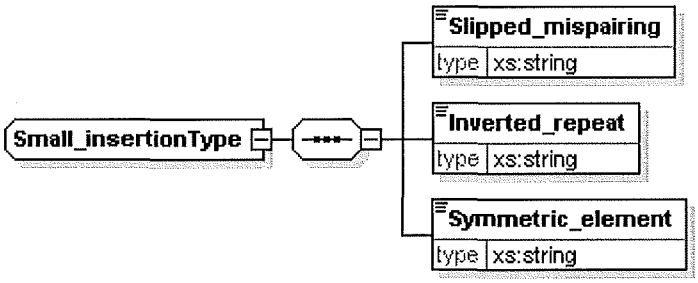
```
<xs:complexType name="Single_base_pair_substitutionType">
  <xs:sequence>
    <xs:element name="Coding_region" type="xs:string"/>
    <xs:element name="Splice_sites" type="xs:string"/>
    <xs:element name="Promoter_region" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
```

element Single_base_pair_substitutionType/Coding_region diagram
namespace http://my_pathological_lesions.com/namespace
type xs:string
source <xs:element name="Coding_region" type="xs:string"/>

element Single_base_pair_substitutionType/Splice_sites diagram
namespace http://my_pathological_lesions.com/namespace
type xs:string
source <xs:element name="Splice_sites" type="xs:string"/>

element Single_base_pair_substitutionType/Promoter_region diagram
namespace http://my_pathological_lesions.com/namespace
type xs:string
source <xs:element name="Promoter_region" type="xs:string"/>

complexType Small_insertionType



namespace http://my_pathological_lesions.com/namespace
children Slipped_mispairing Inverted_repeat Symmetric_element
used by element InsertionType/Small_insertion
source

```
<xs:complexType name="Small_insertionType">
 <xs:sequence>
  <xs:element name="Slipped_mispairing" type="xs:string"/>
  <xs:element name="Inverted_repeat" type="xs:string"/>
  <xs:element name="Symmetric_element" type="xs:string"/>
 </xs:sequence>
</xs:complexType>
```

element Small_insertionType/Slipped_mispairing diagram
namespace http://my_pathological_lesions.com/namespace
type xs:string
source <xs:element name="Slipped_mispairing" type="xs:string"/>

element Small_insertionType/Inverted_repeat diagram
namespace http://my_pathological_lesions.com/namespace
type xs:string
source <xs:element name="Inverted_repeat" type="xs:string"/>

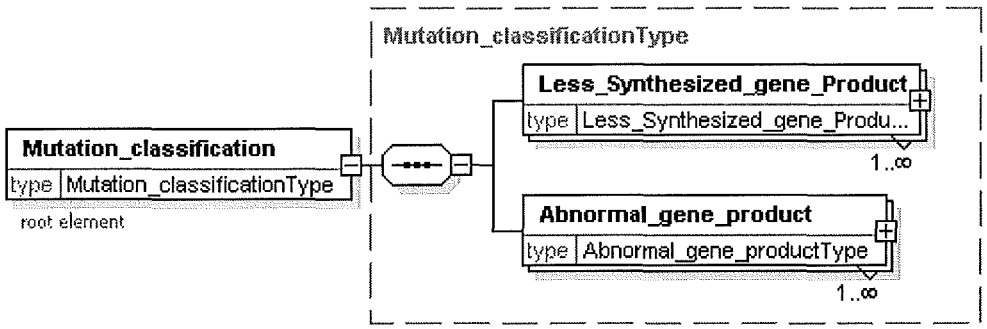
element Small_insertionType/Symmetric_element diagram
namespace http://my_pathological_lesions.com/namespace
type xs:string
source <xs:element name="Symmetric_element" type="xs:string"/>

Schema **MutationClass.xsd**

Elements	**Complex types**
Mutation_classification	Abnormal_gene_productType
Translation_affected	Elongated_gene_productType
	Fused_genesType
	Fused_geneType
	InsertionType
	Less_Synthesized_gene_ProductType
	Mutation_classificationType
	Shortened_gene_product

Small_insertionType
Transcription_affectedType
Translation_affectedType

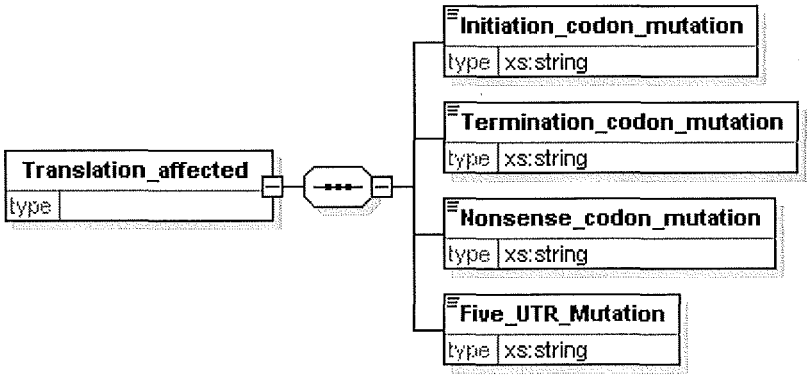
element Mutation_classification



type Mutation_classificationType
children Less_Synthesized_gene_Product, Abnormal_gene_product annotationdocumentation, root element
source

```
<xs:element name="Mutation_classification"
type="Mutation_classificationType">
 <xs:annotation>
  <xs:documentation> root element</xs:documentation>
 </xs:annotation>
</xs:element>
```

element Translation_affected



namespace http://variance_analysis.com/namespace
children Initiation_codon_mutation, Termination_codon_mutation Nonsense_codon_mutation, Five_UTR_Mutation
Source

```
<xs:element name="Translation_affected">
 <xs:complexType>
  <xs:sequence>
   <xs:element name="Initiation_codon_mutation"
       type="xs:string"/>
   <xs:element name="Termination_codon_mutation"
       type="xs:string"/>
   <xs:element name="Nonsense_codon_mutation"
       type="xs:string"/>
   <xs:element name="Five_UTR_Mutation" type="xs:string"/>
  </xs:sequence>
```

```
 </xs:complexType>
</xs:element>
```

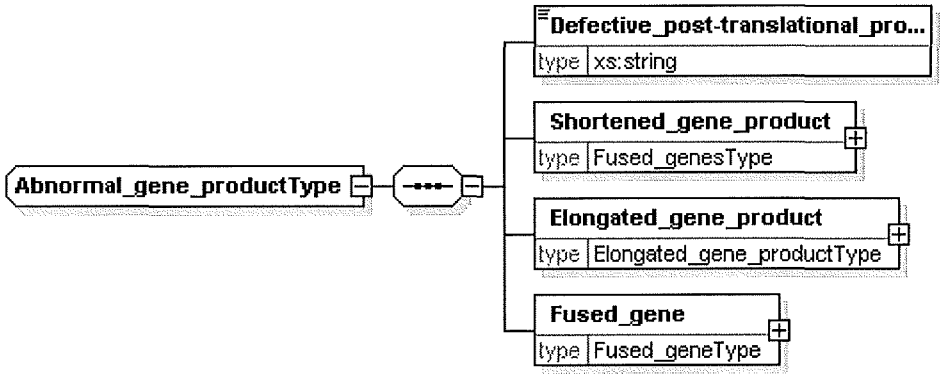
element Translation_affected/Initiation_codon_mutation diagram
type xs:string
source
```
<xs:element name="Initiation_codon_mutation" type="xs:string"/>
```

element Translation_affected/Termination_codon_mutation diagram
type xs:string
source
```
<xs:element name="Termination_codon_mutation"
type="xs:string"/>
```

element Translation_affected/Nonsense_codon_mutation diagram
type xs:string
source
```
<xs:element name="Nonsense_codon_mutation" type="xs:string"/>
```

element Translation_affected/Five_UTR_Mutation diagram
type xs:string
source
```
<xs:element name="Five_UTR_Mutation" type="xs:string"/>
```

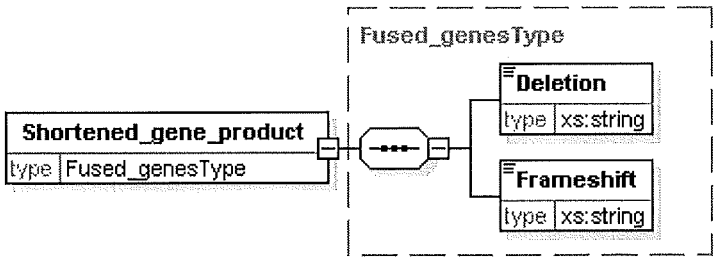
complexType Abnormal_gene_productType



children Defective_post-translational_processing, Shortened_gene_product, Elongated_gene_product, Fused_gene
used by element Mutation_classificationType/Abnormal_gene_product
source
```
<xs:complexType name="Abnormal_gene_productType">
 <xs:sequence>
  <xs:element name="Defective_post-translational_processing"
       type="xs:string"/>
  <xs:element name="Shortened_gene_product"
       type="Fused_genesType"/>
  <xs:element name="Elongated_gene_product"
       type="Elongated_gene_productType"/>
  <xs:element name="Fused_gene" type="Fused_geneType"/>
 </xs:sequence>
</xs:complexType>
```

element Abnormal_gene_productType/Defective_post-translational_processing
type xs:string
source
```
<xs:element name="Defective_post-translational_processing"
type="xs:string"/>
```

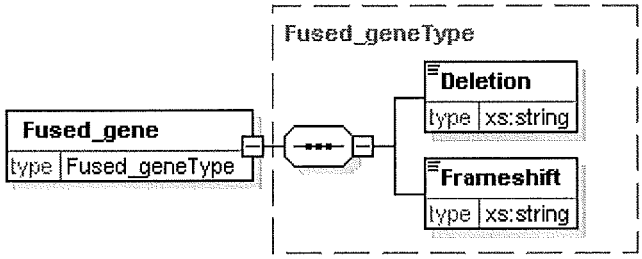
element Abnormal_gene_productType/Shortened_gene_product



type Fused_genesType
children Deletion Frameshift
source <xs:element name="Shortened_gene_product"
type="Fused_genesType"/>

element Abnormal_gene_productType/Elongated_gene_product
type Elongated_gene_productType
children Insertion
source <xs:element name="Elongated_gene_product"
type="Elongated_gene_productType"/>

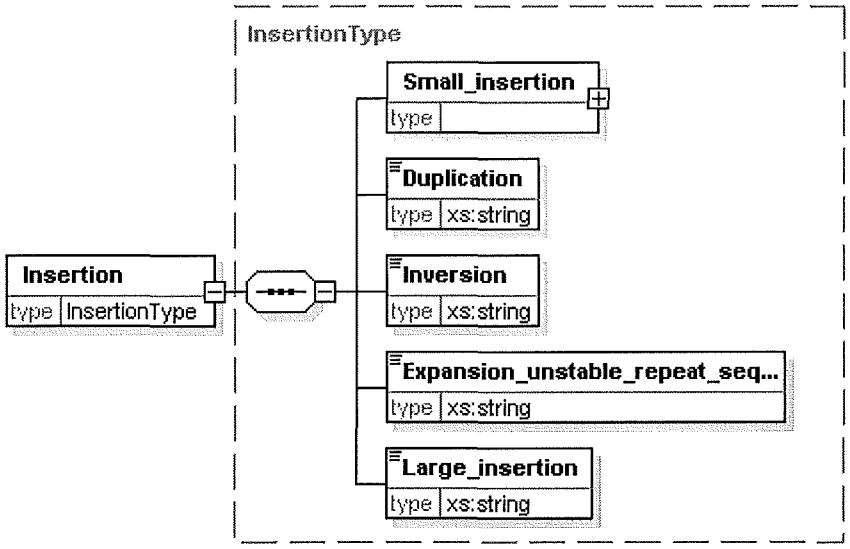
element Abnormal_gene_productType/Fused_gene



type Fused_geneType
children Deletion Frameshift
source <xs:element name="Fused_gene" type="Fused_geneType"/>

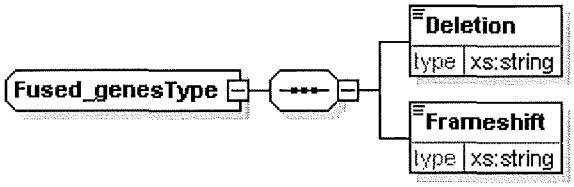
complexType Elongated_gene_productType diagram
children Insertion
used by element Abnormal_gene_productType/Elongated_gene_product
source
```
<xs:complexType name="Elongated_gene_productType">
  <xs:sequence>
    <xs:element name="Insertion" type="InsertionType"/>
  </xs:sequence>
</xs:complexType>
```

element Elongated_gene_productType/Insertion



type InsertionType
children Small_insertion, Duplication Inversion, Expansion_unstable_repeat_sequence, Large_insertion
Source <xs:element name="Insertion" type="InsertionType"/>

complexType Fused_genesType



children Deletion, Frameshift
used by element Abnormal_gene_productType/Shortened_gene_product
complexType Shortened_gene_product
source

```
<xs:complexType name="Fused_genesType">
  <xs:sequence>
    <xs:element name="Deletion" type="xs:string"/>
    <xs:element name="Frameshift" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
```

element Fused_genesType/Deletion diagram
type xs:string
source <xs:element name="Deletion" type="xs:string"/>

element Fused_genesType/Frameshift diagram
type xs:string
source <xs:element name="Frameshift" type="xs:string"/>

complexType Fused_geneType diagram
children Deletion Frameshift
used by element Abnormal_gene_productType/Fused_gene
source

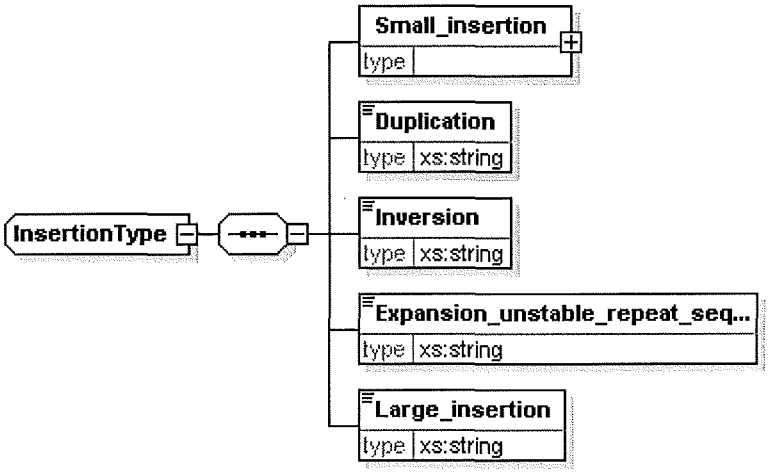
```
<xs:complexType name="Fused_geneType">
  <xs:sequence>
    <xs:element name="Deletion" type="xs:string"/>
```

```
    <xs:element name="Frameshift" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
```

element Fused_geneType/Deletion diagram
type xs:string
source <xs:element name="Deletion" type="xs:string"/>

element Fused_geneType/Frameshift diagram
type xs:string
source <xs:element name="Frameshift" type="xs:string"/>

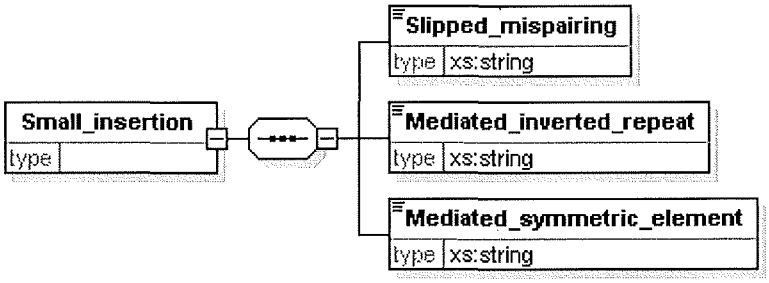
complexType InsertionType



children Small_insertion, Duplication Inversion
Expansion_unstable_repeat_sequence, Large_insertion
used by element Elongated_gene_productType/Insertion
source

```
<xs:complexType name="InsertionType">
 <xs:sequence>
  <xs:element name="Small_insertion">
   <xs:complexType>
    <xs:sequence>
     <xs:element name="Slipped_mispairing" type="xs:string"/>
       <xs:element name="Mediated_inverted_repeat"
       type="xs:string"/>
       <xs:element name="Mediated_symmetric_element"
       type="xs:string"/>
    </xs:sequence>
   </xs:complexType>
  </xs:element>
  <xs:element name="Duplication" type="xs:string"/>
  <xs:element name="Inversion" type="xs:string"/>
        <xs:element name="Expansion_unstable_repeat_sequence"
        type="xs:string"/>
        <xs:element  name="Large_insertion"  type="xs:string"/>
</xs:sequence>
</xs:complexType>
```

element InsertionType/Small_insertion



children Slipped_mispairing, Mediated_inverted_repeat
Mediated_symmetric_element

Source

```
<xs:element name="Small_insertion">
 <xs:complexType>
  <xs:sequence>
   <xs:element name="Slipped_mispairing" type="xs:string"/>
   <xs:element name="Mediated_inverted_repeat"
       type="xs:string"/>
   <xs:element name="Mediated_symmetric_element"
       type="xs:string"/>
  </xs:sequence>
 </xs:complexType>
</xs:element>
```

element InsertionType/Small_insertion/Slipped_mispairing diagram
type xs:string
source <xs:element name="Slipped_mispairing" type="xs:string"/>

element InsertionType/Small_insertion/Mediated_inverted_repeat diagram
type xs:string
source <xs:element name="Mediated_inverted_repeat" type="xs:string"/>

element InsertionType/Small_insertion/Mediated_symmetric_element
type xs:string
source <xs:element name="Mediated_symmetric_element"
type="xs:string"/>

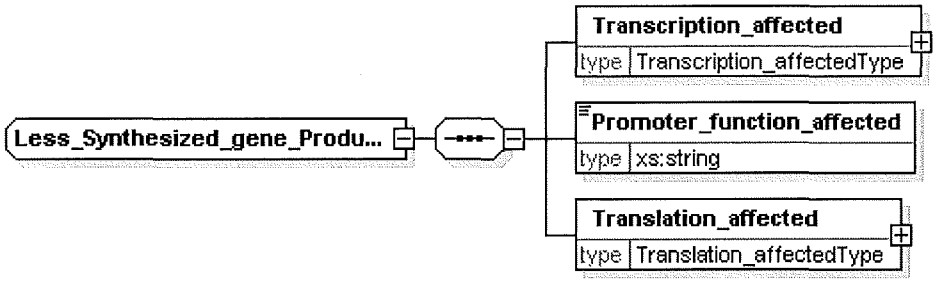
element InsertionType/Duplication diagram
type xs:string
source <xs:element name="Duplication" type="xs:string"/>

element InsertionType/Inversion diagram
type xs:string
source <xs:element name="Inversion" type="xs:string"/>

element InsertionType/Expansion_unstable_repeat_sequence diagram
type xs:string
source <xs:element name="Expansion_unstable_repeat_sequence"
type="xs:string"/>

element InsertionType/Large_insertion diagram
type xs:string
source <xs:element name="Large_insertion" type="xs:string"/>

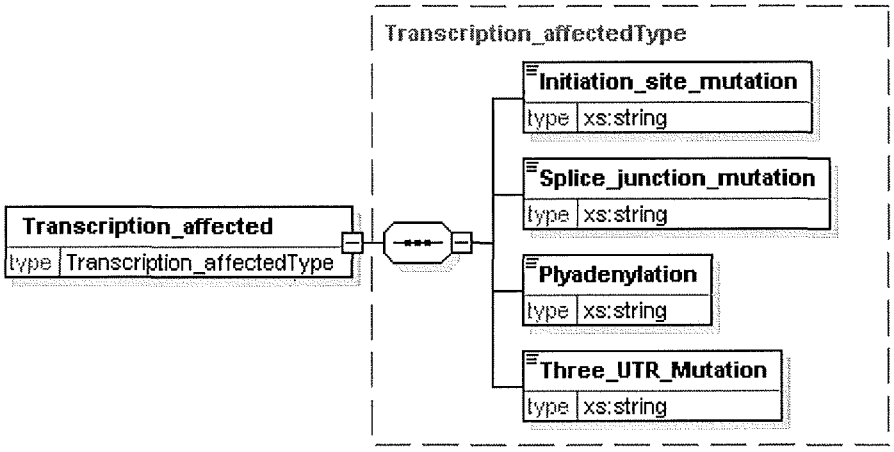
complexType Less_Synthesized_gene_ProductType



children Transcription_affected, Promoter_function_affected , Translation_affected
used by element Mutation_classificationType/Less_Synthesized_gene_Product
source

```
<xs:complexType name="Less_Synthesized_gene_ProductType">
 <xs:sequence>
  <xs:element name="Transcription_affected"
        type="Transcription_affectedType"/>
  <xs:element name="Promoter_function_affected" type="xs:string"/>
  <xs:element name="Translation_affected"
        type="Translation_affectedType"/>
 </xs:sequence>
</xs:complexType>
```

element Less_Synthesized_gene_ProductType/Transcription_affected



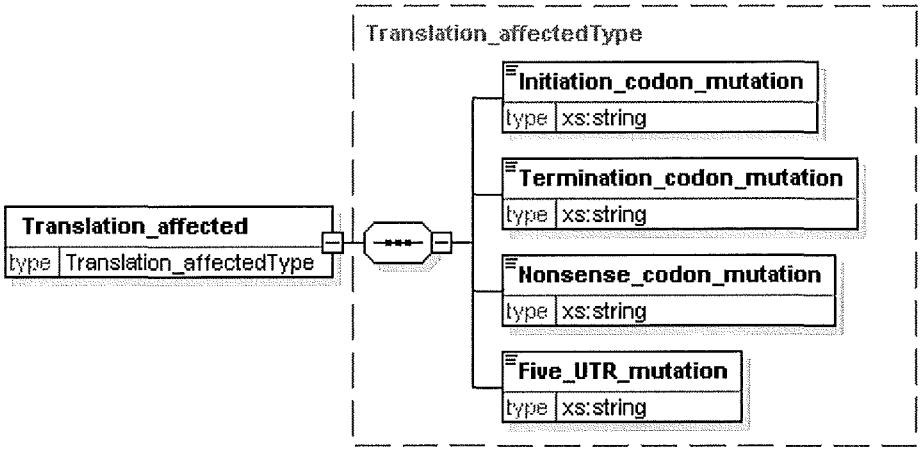
type Transcription_affectedType
children Initiation_site_mutation, Splice_junction_mutation, Plyadenylation Three_UTR_Mutation
Source

```
<xs:element name="Transcription_affected"
type="Transcription_affectedType"/>
```

element Less_Synthesized_gene_ProductType/Promoter_function_affected
type xs:string
source

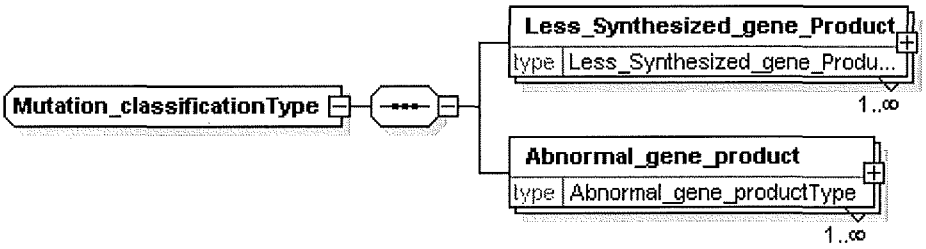
```
<xs:element name="Promoter_function_affected" type="xs:string"/>
```

element Less_Synthesized_gene_ProductType/Translation_affected



type Translation_affectedType
children Initiation_codon_mutation, Termination_codon_mutation, Nonsense_codon_mutation, Five_UTR_mutation
Source <xs:element name="Translation_affected" type="Translation_affectedType"/>

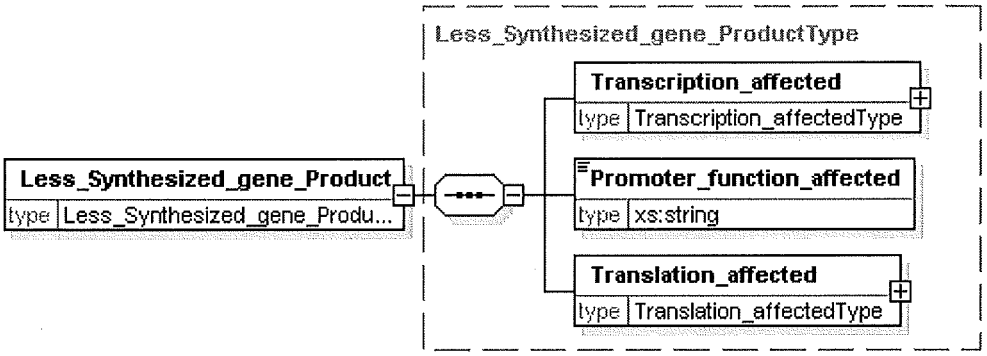
complexType Mutation_classificationType



children Less_Synthesized_gene_Product Abnormal_gene_product
used by element Mutation_classification
source

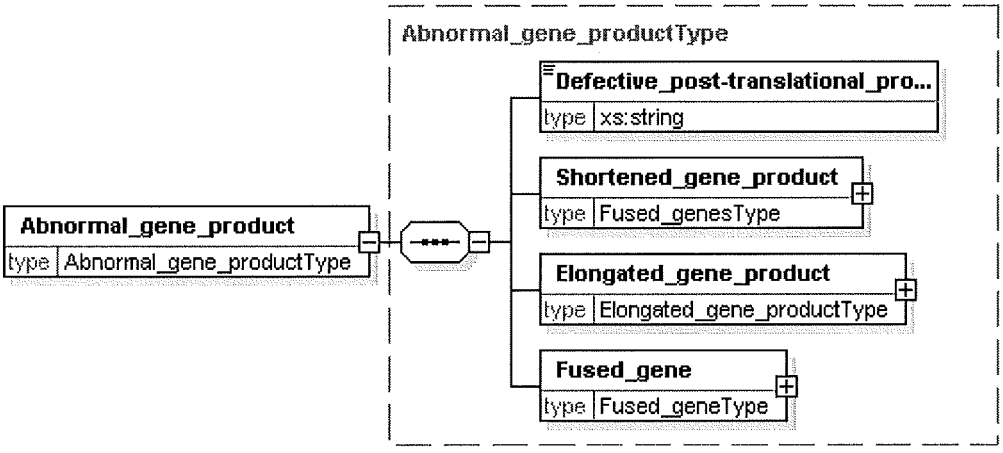
```
<xs:complexType name="Mutation_classificationType">
 <xs:sequence>
  <xs:element name="Less_Synthesized_gene_Product"
        type="Less_Synthesized_gene_ProductType"
        maxOccurs="unbounded"/>
   <xs:element name="Abnormal_gene_product"
   type="Abnormal_gene_productType" maxOccurs="unbounded"/>
 </xs:sequence>
</xs:complexType>
```

element Mutation_classificationType/Less_Synthesized_gene_Product



type Less_Synthesized_gene_ProductType
children Transcription_affected, Promoter_function_affected, Translation_affected
Source <xs:element name="Less_Synthesized_gene_Product" type="Less_Synthesized_gene_ProductType" maxOccurs="unbounded"/>

element Mutation_classificationType/Abnormal_gene_product

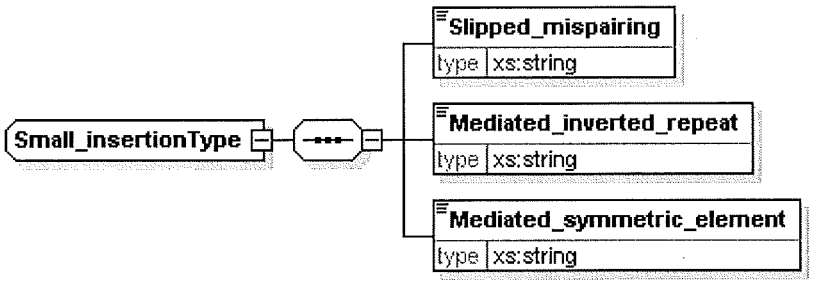


type Abnormal_gene_productType
children Defective_post-translational_processing, Shortened_gene_product , Elongated_gene_product, Fused_gene
Source <xs:element name="Abnormal_gene_product" type="Abnormal_gene_productType" maxOccurs="unbounded"/>

complexType Shortened_gene_product diagram
type extension of Fused_genesType
children Deletion, Frameshift
source

```
<xs:complexType name="Shortened_gene_product">
  <xs:complexContent>
    <xs:extension base="Fused_genesType"/>
  </xs:complexContent>
</xs:complexType>
```

complexType Small_insertionType



children Slipped_mispairing, Mediated_inverted_repeat, Mediated_symmetric_element

Source

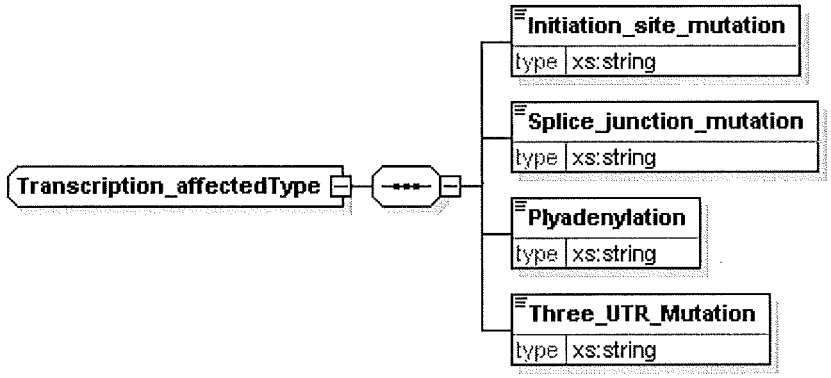
```
<xs:complexType name="Small_insertionType">
 <xs:sequence>
  <xs:element name="Slipped_mispairing" type="xs:string"/>
  <xs:element name="Mediated_inverted_repeat"
        type="xs:string"/>
  <xs:element name="Mediated_symmetric_element"
        type="xs:string"/>
 </xs:sequence>
</xs:complexType>
```

element Small_insertionType/Slipped_mispairing diagram
type xs:string
source <xs:element name="Slipped_mispairing" type="xs:string"/>

element Small_insertionType/Mediated_inverted_repeat diagram
type xs:string
source <xs:element name="Mediated_inverted_repeat" type="xs:string"/>

element Small_insertionType/Mediated_symmetric_element diagram
type xs:string
source <xs:element name="Mediated_symmetric_element"
type="xs:string"/>

complexType Transcription_affectedType



children Initiation_site_mutation, Splice_junction_mutation, Plyadenylation, Three_UTR_Mutation

used by element Less_Synthesized_gene_ProductType/Transcription_affected

source

```
<xs:complexType name="Transcription_affectedType">
 <xs:sequence>
  <xs:element name="Initiation_site_mutation" type="xs:string"/>
```

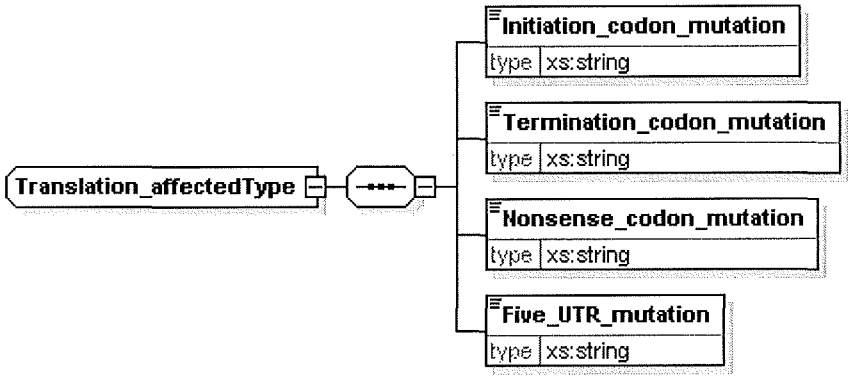
```
    <xs:element name="Splice_junction_mutation"
        type="xs:string"/>
    <xs:element name="Plyadenylation" type="xs:string"/>
    <xs:element name="Three_UTR_Mutation" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
```

element Transcription_affectedType/Initiation_site_mutation diagram
type xs:string
source <xs:element name="Initiation_site_mutation" type="xs:string"/>
element Transcription_affectedType/Splice_junction_mutation diagram
type xs:string
source <xs:element name="Splice_junction_mutation" type="xs:string"/>

element Transcription_affectedType/Plyadenylation diagram
type xs:string
source <xs:element name="Plyadenylation" type="xs:string"/>

element Transcription_affectedType/Three_UTR_Mutation diagram
type xs:string
source <xs:element name="Three_UTR_Mutation" type="xs:string"/>

complexType Translation_affectedType



children Initiation_codon_mutation, Termination_codon_mutation, Nonsense_codon_mutation, Five_UTR_mutation
used by element Less_Synthesized_gene_ProductType/Translation_affected
source

```
<xs:complexType name="Translation_affectedType">
  <xs:sequence>
    <xs:element name="Initiation_codon_mutation"
        type="xs:string"/>
    <xs:element name="Termination_codon_mutation"
        type="xs:string"/>
    <xs:element name="Nonsense_codon_mutation"
        type="xs:string"/>
    <xs:element name="Five_UTR_mutation" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
```

element Translation_affectedType/Initiation_codon_mutation diagram
type xs:string
source <xs:element name="Initiation_codon_mutation" type="xs:string"/>

element Translation_affectedType/Termination_codon_mutation diagram
type xs:string
source <xs:element name="Termination_codon_mutation" type="xs:string"/>

element Translation_affectedType/Nonsense_codon_mutation diagram
type xs:string
source <xs:element name="Nonsense_codon_mutation" type="xs:string"/>

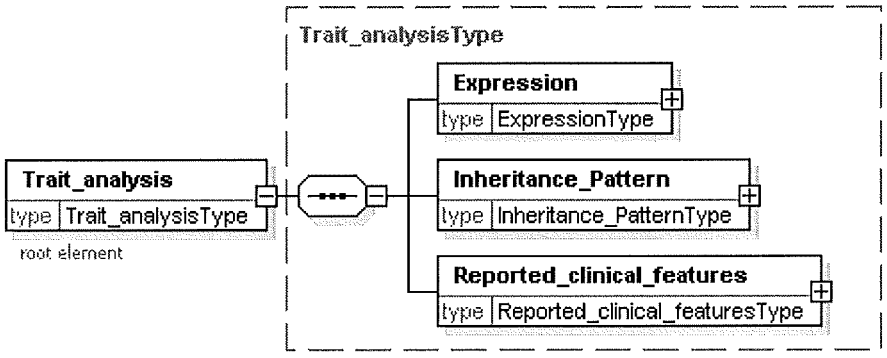
element Translation_affectedType/Five_UTR_mutation diagram
type xs:string
source <xs:element name="Five_UTR_mutation" type="xs:string"/>

Schema **TraitAnalysis.xsd**

Elements
Trait_analysis

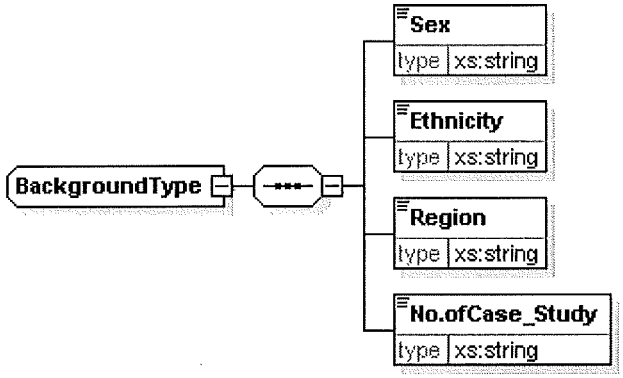
Complex types
BackgroundType
ExpressionType
Genotypic_MeasurementType
Inheritance_PatternType
Phenotypic_measurementType
Reported_clinical_featuresType
Trait_analysisType

element **Trait_analysis**



namespace http://my_trait_analysis.com//namespace
type Trait_analysisType
children Expression, Inheritance_Pattern, Reported_clinical_features
annotationdocumentation root element
source
```
<xs:element name="Trait_analysis" type="Trait_analysisType">
  <xs:annotation>
    <xs:documentation> root element</xs:documentation>
  </xs:annotation>
</xs:element>
```

complexType **BackgroundType**



namespace http://my_trait_analysis.com//namespace
children Sex Ethnicity Region No.ofCase_Study
used by element Inheritance_PatternType/Background
source

```
<xs:complexType name="BackgroundType">
 <xs:sequence>
  <xs:element name="Sex" type="xs:string"/>
  <xs:element name="Ethnicity" type="xs:string"/>
  <xs:element name="Region" type="xs:string"/>
  <xs:element name="No.ofCase_Study" type="xs:string"/>
 </xs:sequence>
</xs:complexType>
```

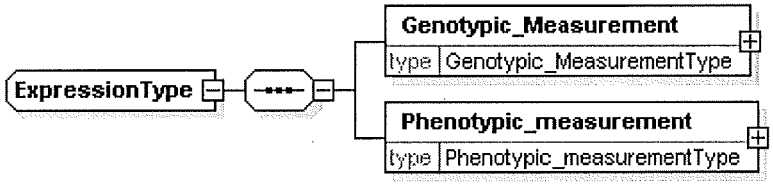
element **BackgroundType/Sex diagram**
namespace http://my_trait_analysis.com//namespace
type xs:string
source <xs:element name="Sex" type="xs:string"/>

element **BackgroundType/Ethnicity diagram**
namespace http://my_trait_analysis.com//namespace
type xs:string
source <xs:element name="Ethnicity" type="xs:string"/>

element **BackgroundType/Region diagram**
namespace http://my_trait_analysis.com//namespace
type xs:string
source <xs:element name="Region" type="xs:string"/>

element **BackgroundType/No.ofCase_Study diagram**
namespace http://my_trait_analysis.com//namespace
type xs:string
source <xs:element name="No.ofCase_Study" type="xs:string"/>

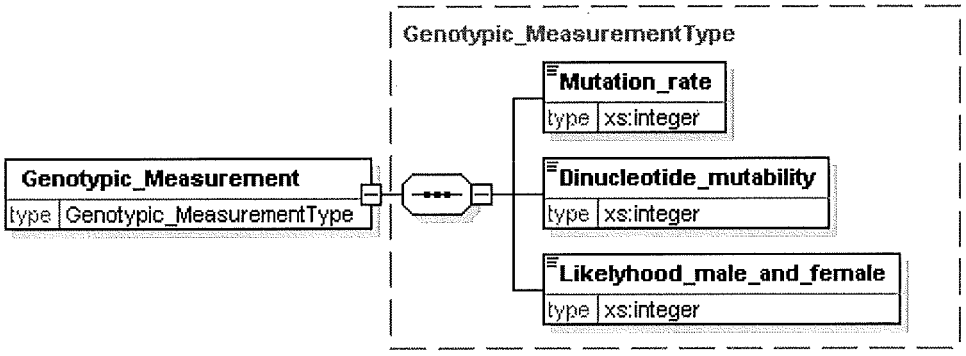
complexType **ExpressionType**



namespace http://my_trait_analysis.com//namespace
children Genotypic_Measurement Phenotypic_measurement
used by element Trait_analysisType/Expression
source

```
<xs:complexType name="ExpressionType">
  <xs:sequence>
    <xs:element name="Genotypic_Measurement"
          type="Genotypic_MeasurementType"/>
    <xs:element name="Phenotypic_measurement"
          type="Phenotypic_measurementType"/>
  </xs:sequence>
</xs:complexType>
```

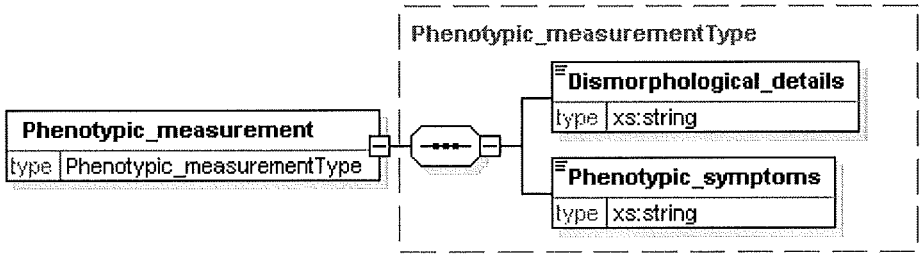
element **ExpressionType/Genotypic_Measurement**



namespace http://my_trait_analysis.com//namespace
type Genotypic_MeasurementType
childrenMutation_rate Dinucleotide_mutability
Likelyhood_male_and_female
Source

```
<xs:element name="Genotypic_Measurement"
type="Genotypic_MeasurementType"/>
```

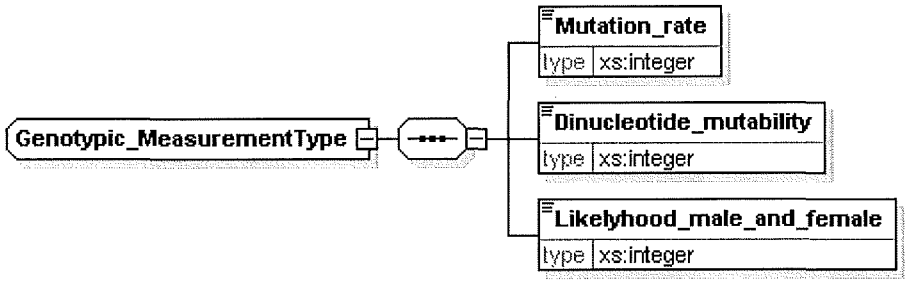
element **ExpressionType/Phenotypic_measurement**



namespace http://my_trait_analysis.com//namespace
type Phenotypic_measurementType
children Dismorphological_details Phenotypic_symptoms
source

```
<xs:element name="Phenotypic_measurement"
type="Phenotypic_measurementType"/>
```

complexType **Genotypic_MeasurementType**



namespace http://my_trait_analysis.com//namespace
children Mutation_rate Dinucleotide_mutability Likelyhood_male_and_female
used by element ExpressionType/Genotypic_Measurement
source

```
<xs:complexType name="Genotypic_MeasurementType">
 <xs:sequence>
  <xs:element name="Mutation_rate" type="xs:integer"/>
  <xs:element name="Dinucleotide_mutability" type="xs:integer"/>
  <xs:element name="Likelyhood_male_and_female"
       type="xs:integer"/>
 </xs:sequence>
</xs:complexType>
```

element **Genotypic_MeasurementType/Mutation_rate diagram**
namespace http://my_trait_analysis.com//namespace
type xs:integer
source

```
<xs:element name="Mutation_rate" type="xs:integer"/>
```

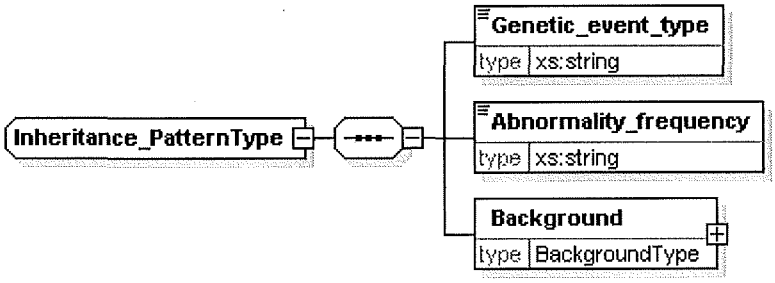
element **Genotypic_MeasurementType/Dinucleotide_mutability diagram**
namespace http://my_trait_analysis.com//namespace
type xs:integer
source

```
<xs:element name="Dinucleotide_mutability" type="xs:integer"/>
```

element **Genotypic_MeasurementType/Likelyhood_male_and_female**
namespace http://my_trait_analysis.com//namespace
type xs:integer
source

```
<xs:element name="Likelyhood_male_and_female"
       type="xs:integer"/>
```

complexType **Inheritance_PatternType**



namespace http://my_trait_analysis.com//namespace
children Genetic_event_type Abnormality_frequency Background
used by element Trait_analysisType/Inheritance_Pattern
source

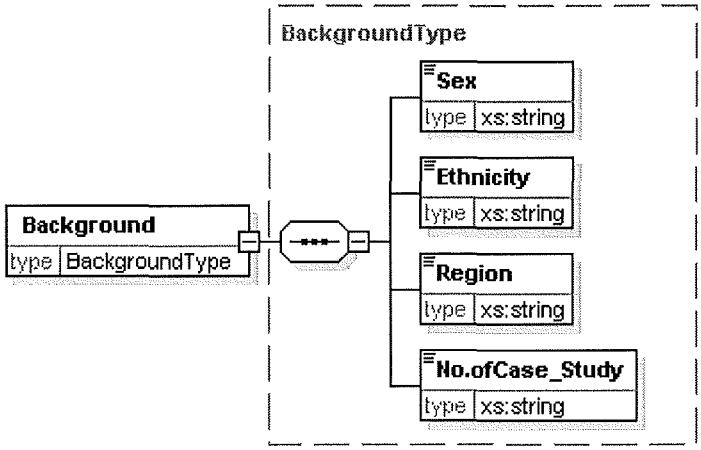
```
<xs:complexType name="Inheritance_PatternType">
 <xs:sequence>
```

```
    <xs:element name="Genetic_event_type" type="xs:string"/>
    <xs:element name="Abnormality_frequency" type="xs:string"/>
    <xs:element name="Background" type="BackgroundType"/>
  </xs:sequence>
</xs:complexType>
```

element **Inheritance_PatternType/Genetic_event_type**
namespace http://my_trait_analysis.com//namespace
type xs:string
source <xs:element name="Genetic_event_type" type="xs:string"/>

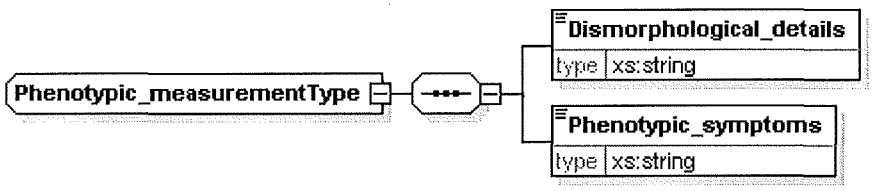
element **Inheritance_PatternType/Abnormality_frequency diagram**
namespace http://my_trait_analysis.com//namespace
type xs:string
source <xs:element name="Abnormality_frequency" type="xs:string"/>

element **Inheritance_PatternType/Background**



namespace http://my_trait_analysis.com//namespace
type BackgroundType
children Sex Ethnicity Region No.ofCase_Study
source <xs:element name="Background" type="BackgroundType"/>

complexType **Phenotypic_measurementType**



namespace http://my_trait_analysis.com//namespace
children Dismorphological_details Phenotypic_symptoms
used by element ExpressionType/Phenotypic_measurement
source

```
<xs:complexType name="Phenotypic_measurementType">
  <xs:sequence>
    <xs:element name="Dismorphological_details"
          type="xs:string"/>
    <xs:element name="Phenotypic_symptoms" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
```

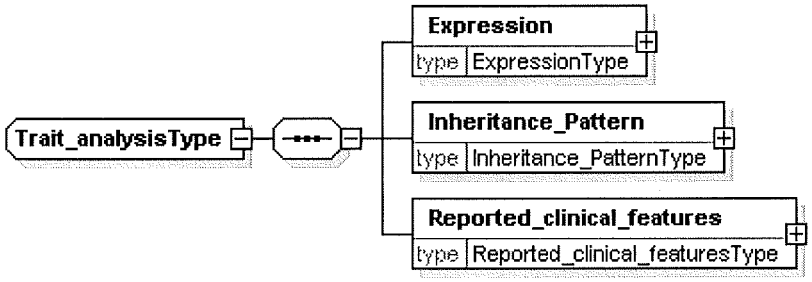
element **Phenotypic_measurementType/Dismorphological_details**
namespace http://my_trait_analysis.com//namespace
type xs:string
source
```
<xs:element name="Dismorphological_details" type="xs:string"/>
```

element **Phenotypic_measurementType/Phenotypic_symptoms diagram**
namespace http://my_trait_analysis.com//namespace
type xs:string
source
```
<xs:element name="Phenotypic_symptoms" type="xs:string"/>
```

complexType **Reported_clinical_featuresType diagram**
namespace http://my_trait_analysis.com//namespace
children Phenotypic_details
used by element Trait_analysisType/Reported_clinical_features
source
```
<xs:complexType name="Reported_clinical_featuresType">
  <xs:sequence>
    <xs:element name="Phenotypic_details" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
```

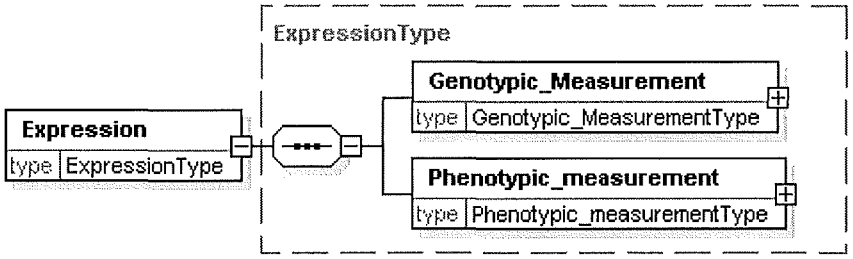
element **Reported_clinical_featuresType/Phenotypic_details diagram**
namespace http://my_trait_analysis.com//namespace
type xs:string
source
```
<xs:element name="Phenotypic_details" type="xs:string"/>
```

complexType **Trait_analysisType**



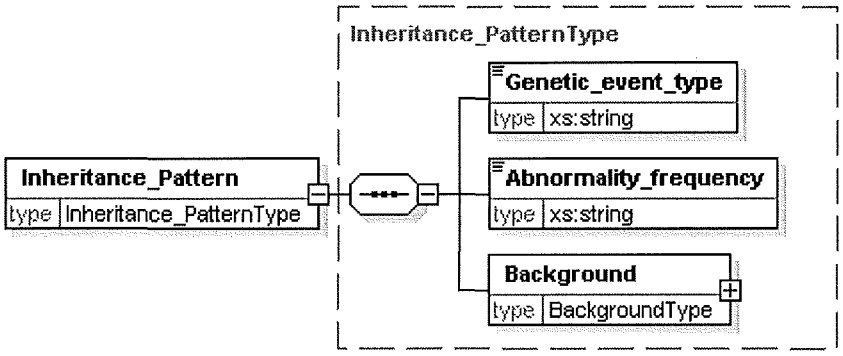
namespace http://my_trait_analysis.com//namespace
children Expression Inheritance_Pattern Reported_clinical_features
used by element Trait_analysis
source
```
<xs:complexType name="Trait_analysisType">
  <xs:sequence>
    <xs:element name="Expression" type="ExpressionType"/>
    <xs:element name="Inheritance_Pattern"
          type="Inheritance_PatternType"/>
    <xs:element name="Reported_clinical_features"
          type="Reported_clinical_featuresType"/>
  </xs:sequence>
</xs:complexType>
```

element **Trait_analysisType/Expression**



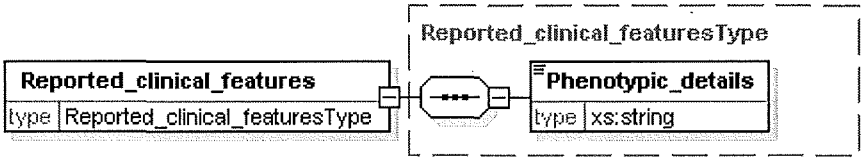
namespace http://my_trait_analysis.com//namespace
type ExpressionType
children Genotypic_Measurement Phenotypic_measurement
source <xs:element name="Expression" type="ExpressionType"/>

element **Trait_analysisType/Inheritance_Pattern**



namespace http://my_trait_analysis.com//namespace
type Inheritance_PatternType
childrenGenetic_event_type Abnormality_frequency Background
source <xs:element name="Inheritance_Pattern"
type="Inheritance_PatternType"/>

element **Trait_analysisType/Reported_clinical_features**



namespace http://my_trait_analysis.com//namespace
type Reported_clinical_featuresType
children Phenotypic_details
source <xs:element name="Reported_clinical_features"
type="Reported_clinical_featuresType"/>

3.4.2 A hierarchy for gene mutation data

The data model of human gene mutation data based on the non-redundant schema integration concept (Khan and Rahman 2001b) is presented in Figure 3.5. This section explains the details of this proposed concept. In this model data have been presented as objects which are instances of classes. We have emphasized on the

domain of mutation data. All the superclasses and subclasses are the representation of this concept. *MutatedGene* class represents any common features of the gene that has been mutated. It will store the coding region, promoter region, consensus sequence and the transcribed area. Two subclasses that are referred by this superclass objects are *DiseaseCausing* class and *IndirectAssociation* class. The *DiseaseCausing* class is further subdivided into two subclasses: *SequenceFeature* and *ProductFeature*. *SequenceFeature* stores the information on mutation site characteristics, *i.e.*, cross-over, and gap. *ProductFeature* stores information on protein structure, affinity site modification and quantity of products. *AssayTechnique* class supports the information with laboratory protocol and empirical information, *i.e.*, images of Radiography, Ideogram *etc.* The object classes described here are independent but the two main classes, *WildtypeGene* and *MutatedGene* are related to each other. This relation has been represented with the arrow in Figure 3.5. The relation itself has been represented as a class, *Mutation* (see Glossary). This relation is functionally dependent on each other, *i.e.*, a mutated gene depends on a particular gene sequence. This *Mutation* class is specialised with two subclasses, *MutationTypes* and *MutationMeasurement*. *MutationTypes* class will categorise the mutation data and it will store the information in two distinctive subclasses. Mutation data is stored either in class *LesssynthesizedGeneProduct* or in *AbnormalGeneProduct*. This depends upon the nature and the characteristics of the mutation. *LesssynthesizedGeneProduct* corresponds to the information if the mutation causes reduced synthesis of a normal gene product. Mutation type could be deletions (frame shifts), insertions, duplication, splice junction mutations, *etc.* *AbnormalGeneProduct* class gathers the information on gene structural defects, *i.e.*, elongated gene product or shortened gene product. It also stores information related to the post translational defects, *i.e.*, modification of processing and instability of protein products. *MutationMeasurement* will store the qualitative and quantitative measurements of the mutation, *i.e.*, Mutation Rate and Mutation from public domain databases and it will depend on the mutation data entry.

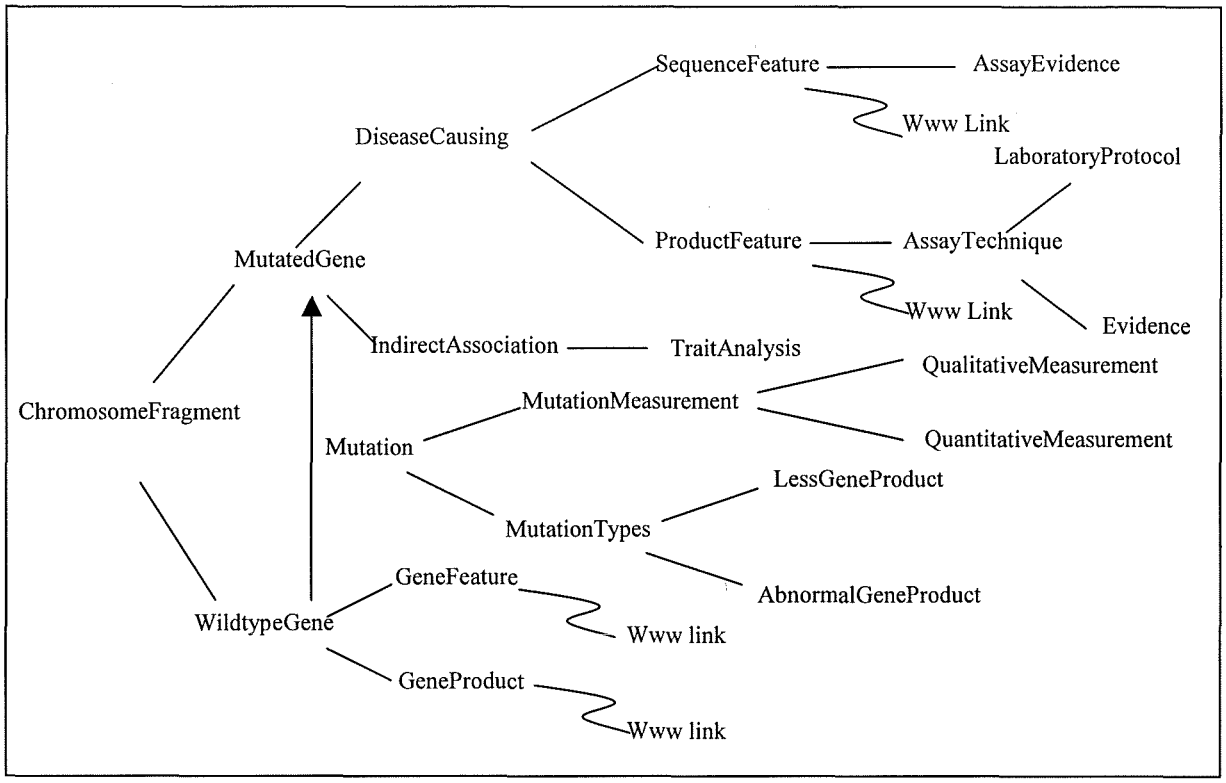


Figure 3.5 The hierarchy used in gene mutation data model

3.4.3 Non-Redundant schema integration

At present the interoperability between databases is achieved by designing generic class and attributes applicable for general community. This results in overlapping of attributes and objects represented by different classes in different databases. For example, *Gene* classes exist in both the Genome Database (GDB) and the Genome Sequence Databases (GSD).

The research focuses on designing a schema for genetic disorder database which will have unique classes, attributes and objects for variance analysis. This will be unique since it will have no overlapping components with other genetic databases. This implies that if attributes a_{mbd1}, a_{mbd2}, a_{mbd3}, ...a_{mbdn} exist in classes C_{mbd1}, C_{mbd2}, C_{mbd3}, ...C_{mbn} of public domain molecular biology databases in schema S_{mbd1}, S_{mbd2}, S_{mbd3}, ...S_{mbdn}, then the same attributes (a_{mbd1}, ...a_{mbdn}) and classes (C_{mbd1}, ...C_{mbdn}) will not exist in the schema of genetic disorder disease database S_{gd}. The intersection of C_{mbd1}, C_{mbd2}, C_{mbd3}, ...C_{mbn} can exist but the intersection of C_{mbd1}, C_{mbdn} with any classes of S_{gd} will not exist (Figure 3.6). The classes, C_{gd1}, ...C_{gdn} which belongs to S_{gd} will be unique within all the heterogeneous databases of scientific interest. The classes C_{gd1}, ..., C_{gdn} can be a subclass of classes C_{mbd1}, C_{mbd2}, C_{mbd3}, ..., C_{mbdn} but

$C_{gd1}, ..., C_{gdn}$ can not be subset of these $C_{mbd1}, ..., C_{mbdn}$ classes. The classes $C_{gd1}, ..., C_{gdn}$ can be a superclass of classes C_{mbd1}, C_{mbd2}, $C_{mbd3}, .., C_{mbdn}$ but there can not be any union of attributes belonging to $C_{mbd1}, ..., C_{mbdn}$ classes (Figure 3.6). This will ensure that C_{gd} can never be a part of any derived subclass or superclass. The attributes of classes $C_{gd1}, ..., C_{gdn}$ will not be a composite attributes by taking the values from the attributes of $C_{mbd1}, ..., C_{mbdn}$ classes. This will ensure no data redundancy during constructing data warehouses and its interoperability between heterogeneous databases. This non-redundant schema integration (Khan and Rahman, 2001b) is summarised here.

i) *Attribute* $\boldsymbol{a_{gd} \not\subset \{b_1,...,b_n\}}$ *of* C_{mbdi} *classes where* $1 \leq i \leq n$ *[since* b_n *is attribute of classes belongs to* S_{mbdi}*]*

ii) *The attributes* a_{gd} *of* C_{gd} *will not have the union of attributes* $\boldsymbol{\{b_1,...,b_n\}}$ *of* C_{mbdi} *classes where* $1 \leq i \leq n$ *[since* b_n *is attribute of classes belongs to* S_{mbdi}*]*

iii) *The attributes* a_{gd} *of* C_{gd} *will not have the intersection of attributes* $\boldsymbol{\{b_1,...,b_n\}}$ *of* C_{mbdi} *classes where* $1 \leq i \leq n$ *[since* b_n *is attribute of classes belongs to* S_{mbdi}*]*

iv) *The attribute* a_{gd} *of class* C_{gdi} *will not be a composition of other attributes like* $b_1[C_{mbd1}]$, $b_2[C_{mbd2}], ..., b_n[C_{mbdn}]$ *where each* C_{mbdn} $(n \geq 1)$ *denotes a class and each* b_n *denotes an attribute associated with* C_{mbdk} *(where* $k=1,2,....,n$*).*

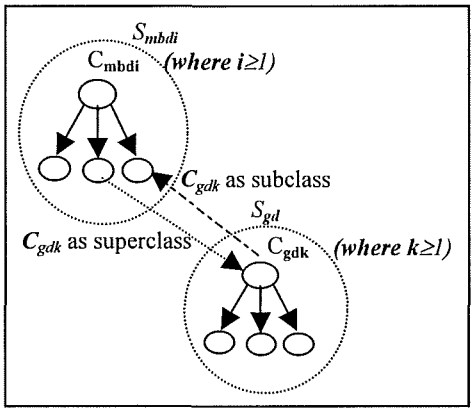


Figure 3.6. Interacting classes for interoperability

(Note: Class C_{gdk} of schema S_{gd} is acting as subclass for interaction with other molecular biology database schema classes and vice versa.)

3.5. Summary

The chapter sets out a formalism for biological resource integration to avoid dependency on global schema and to increase the priority on primary sources of data. It starts with the discussion 'why laboratory based gene disorder database is necessary to store empirical evidence of a particular research laboratory'. For example, we need to store parameters like mutation frequency and mutation rate for variance analysis of the gene mutation data. Image data is needed to analyse the data with other data available in public domain databases like GenBank, GDB and PDB. Although, Human Gene Mutation Database (HGMD) stores the gene mutation data it does not deal with indirect association of diseases. For indirect association of disease analysis it needs to examine the background history, ethnicity, recombination fraction, mutation rate and to determine likelihood of the diseases in human. Hence, it requires mathematical and statistical modelling for analysing the inheritance pattern of diseases. Thus, a detail gene mutation data modelling is required to store all the relevant information. In this context small, dedicated and laboratory based database is much easier to maintain and it will not affect the global transactions if modified. However, as the public domain databases are the rich source of information, the suggested gene disorder data model alone will not be able to examine molecular mechanism of the diseases. It requires to share knowledge with other databases on common view for variance analysis, *i.e.*, protein morphological comparison and gel-electrophoresis image comparison.

The chapter also shows that the data model can not be implemented using traditional approach like relational or object oriented model because of its limitation in handling semistructured data. Molecular biology data modelling needs to interact with semistructured data. Semistructured data modelling language, XML, is used to implement the model for gene mutation data. In this the data models organise the data in graphs where each node represents an object and edges represent the relationship between the objects and values represented by the edge's node. The labels are created so that it can add semantics on the values or the objects.

A hierarchical model is proposed to describe the position of the component database in a network environment so that element or object redundancy can be avoided. A formal description to avoid element or object redundancy and to eliminate

those attributes which are already present in the public domain databases is suggested in this chapter. The framework focuses on the concept of describing the component data model without having overlapping attributes. It also focuses on the criteria where the task-specific dedicated database (genetic disorder database) does not hold any subset or union of attributes that are currently present in public domain databases. It does not also hold any attributes which are intersection of existing attributes in the public domain databases. Thus, by interacting with the *parent class/object* and *child class/object* of public domain databases, interoperability among the resource databases can be achieved.

The approaches described in this chapter establishes a theoretical foundation of the 'molecular biology database integration'. These theoretical concepts in contrast to conventional methods, *e.g.*, data warehousing or database linking using hyper text, include values of integrating data. It presents compound results by using single interface for variance analysis of gene mutation data without any data redundancy and by incorporating related entities of data sources.

The next chapter discusses how laboratory evidence which is stored in component database of the laboratory, such as gel electrophoresis image, can initiate the integration process for data extraction from public domain databases.

Chapter 4

Identifying Protein Spots in 2D Gel Electrophoresis Images

Chapter Objective

Many algorithms are available for quantitative and qualitative analysis of protein spot in gel electrophoresis images and majority of these algorithms use geometric and image processing techniques to match the protein spots. These algorithms do not take into consideration the electrophoretic mobility of the proteins and they only match similar protein spots rather than matching similar proteins. This chapter explores the existing methodologies and approaches for protein spots matching and prepares a ground to set out a new approach to cope with the diversity of protein spots. The chapter also examines the existing method of linking protein spots with phenotypic details and it highlights the drawbacks of the existing method. The chapter also looks at the issues that need to be addressed in the new method.

Chapter Contents

Chapter 4

Identifying Protein Spots in 2D Gel Electrophoresis Images

4.1. Introduction

Gel electrophoresis technique is a fundamental procedure for separating the DNA and proteins from a mixture. This technique is based on the concept of separating the charged particles in the gel. Gel electrophoresis analysis is a well-established technique to compare and contrast one protein with another. It uses an electric field to separate a mixture of molecules through a stationary material (gel). The velocity of migration of a protein in an electric field depends on the electric field strength, the net charge on the protein and the frictional coefficient (see Glossary). The electric force drives the charged molecules towards the charged electrode. There is also a frictional resistance that slows down the movement of this charged molecule. This frictional force is a measure of the *hydrodynamic size* of the molecule, the shape of the molecule, the pore size of the medium in which electrophoresis takes place, and the viscosity of the buffer.

In electrophoresis, the force moving the macromolecule (nucleic acids or proteins) is the electrical potential and the electrophoretic mobility of an ion is the ratio of the velocity of the particle to the electrical potential.

When a potential difference is applied, molecules with different overall charges will begin to separate due to their different electrophoretic mobilities. Even molecules with similar charges will begin to separate if they have different molecular sizes, since they will experience different frictional forces. Some methods of electrophoresis rely on the different charges applied on molecules to effect separation, while other methods exploit differences in molecular size which therefore encourage frictional effects to bring about separation. As the separation process continues, the separation between the larger and the smaller fragments increases. The schematic diagram of gel electrophoresis image is shown in Figure 4.1.

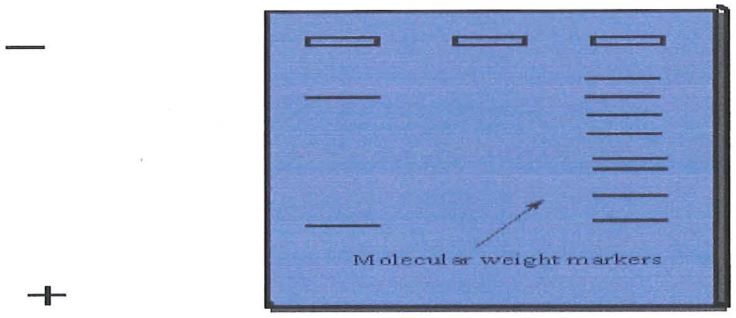


Figure 4.1. A schematic diagram of gel electrophoresis technique

Molecules that are small compared with other proteins in the gel readily move through the gel to the positive charge, whereas molecules much larger than the other proteins are almost immobile. Intermediate size molecules move through the gel with various degrees of mobility. The migrated band of the proteins are then compared with a set of markers to determine the molecular weight of the protein.

A gel electrophoresis image which has been used to separate the proteins present in *Erythroleukemic* (see Glossary) cell is shown in Figure 4.2. This is used to analyse the protein components of cells for any genotype or phenotype expression. The 2D gel is the product of two separations performed sequentially in acrylamide gel media. Based on isoelectric charge of the protein considered as the first dimension, and molecular weight of the protein considered as the second dimension, the protein molecules have been separated on the gel medium. Each spot in the 2D pattern of spots of the image represents a protein. The coordinate of any spot is determined by its corresponding isoelectric charge and the molecular weight (Figure 4.3a). This shows that, all the proteins with the same molecular weight will have the same y value although their electric charge (x value) may vary. So all the spots on the same line of path are considered as similar protein with respect to its source protein spot (Figure 4.3b). These spots are detected by staining or radiographic methods. Few spots in the image are identified with protein names, *e.g.*, *enoyl-Coa hydratase* or *triosophosphate isomerase*. These images generate huge qualitative and quantitative data which need to be correlated with the disease states of the human cell for genotype or phenotype explanation. This would then aid to design the relevant drug for the particular disease. For such application, the gel electrophoresis images need to be compared with other images which are available in public domain databases that can be accessed through the internet.

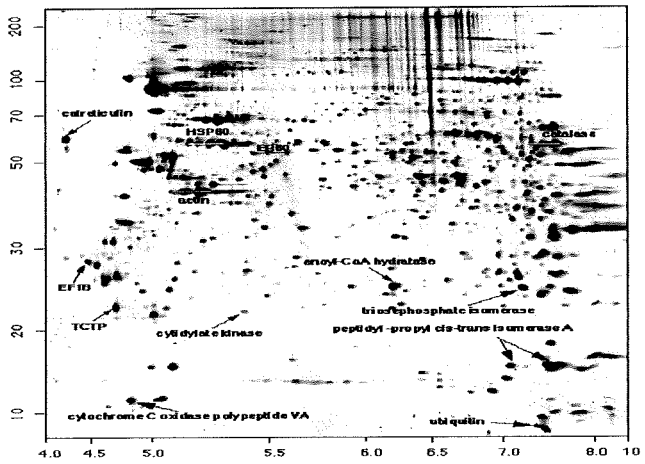


Figure 4.2 Gel electrophoresis image of human Erythroleukemic cell proteins

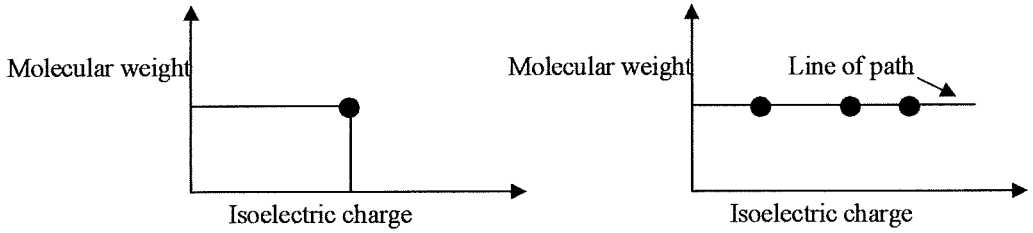


Figure 4.3. (a) a protein spot coordination; (b) Protein spots on the line of path with same molecular weight

However, the dynamic nature of the protein spots makes it a major challenge to compare the protein spots and to search for the identical or similar proteins from the gel electrophoresis images. These type of images have the following characteristics:

- spots in the image determine the protein
- size of the spots vary and they do not have any consistency
- spot features are not fixed for any particular protein
- the electrophoretic mobility (Section 4.1) and isoelectric charge determine the type of protein.

These varying characteristic features raise questions as how to match these with other proteins of similar natures that are available from various resource databases. Section 4.2 reviews the existing algorithms and systems which have been used for matching gel electrophoresis images. The limitations of the existing approaches and the need for alternative approaches to match gel electrophoresis image have been highlighted in this section.

Section 4.3 investigates the spot identification techniques and its applications. It highlights the drawback and weakness of these techniques. The section then describes the proposed spot identification technique which identifies the spot on a specific line of path and the result is then used for matching and finding the correct protein spot. Section 4.3.2 introduces Hoursdorff distance technique which is used to determine the geometric patterns and shapes of the spots in the neighbourhood area of target positional vector. The technique is elaborated further in section 4.3.3. Section 4.4 describes an approach to link the selected and matched protein spot with the relevant 3D structure of it. This technique automates the manual process which is currently used to link the protein spots with the 3D image of the protein. The research shows why such dynamic linking of protein spot with 3D images is necessary for clinical analysis and selecting the right drug. Finally, section 4.5 summarises the limitations of present approaches and highlights the motivation for this alternative and novel approach to compare protein spots.

4.2 Present Software and Algorithms

There are many software available which are used to compare the protein spots. For example, GELLAB system (Lemkin and Lipkin, 1981ab and Lemkin *et al.*, 1982), uses the point pattern comparison technique to identify the spots of the same protein. Another system is MELANIE (Appel *et al.*, 1997) which compares spot clusters instead of using simplified point pattern matching technique. Melanie and Java based program Flicker (Lemkin, 1997) used user defined landmarks for comparison of the spots. It requires alignment assumptions for image comparison. The system also defines a probabilistic criterion for the definition of a correct match (Appel *et al.* 1997). These systems work well for the same type of gel electrophoregram where the intensity of the spots do not change. However in both cases if the intensity value of the spots change then it can significantly affect the outcome of the gel image matching (Kriegel *et al.* 2000 and Chui and Rangaranjan, 2000). Smilansky (2001) concluded that one of the key problems in gel electrophoresis image analysis is the difficulty and slowness in the registration process (image alignment) which is laborious and the results are not satisfactory. He developed a system for the analysis of 2-D PAGE images, called Z3. The Z3 system employs registration of the images from the raw image data as the first step and the subsequent processing of the image then relies on it. He argued that it is an alternative to the customary approach where a spot is detected first in the image and then the

landmarking or the image registration process is applied to compare it further with other images. Spot detection in the first phase is very crucial as it reduces the amount of data from millions of pixel to thousands but adequate information for image registration is not available from public domain databases throughout internet. The shape, the intensity and the noise content of the spots vary from one image to another for an identical protein also. So, image registration, as a first step, will only work in the known environment where all the features of the spots are known.

New algorithms for protein spot matching are also emerging. For example, Panek and Vohradsky (1999), introduced a new algorithm to identify protein spots. Their approach use the information from the neighbourhood spots for matching. A syntactic descriptor characterize the spots and the positional similarity is then derived. Pre-processing phase in their technique also leads to image distortion. Pleibner *et al.* (1999) introduced another algorithm, using watershed transformation and Delaunay Triangulation technique, to detect the protein spots. The algorithm deals with the local matching problem of 2D patterns of protein spots from gel electrophoresis images. In their approach a triangulation is constructed over the set of sample points in the source image. They then computed all the locations in the target image where a good matching with the patterns is likely to occur. Finally, they computed the actual patterns that answer their local matching query. This matching algorithm works only for local intensive patterns. However, this is an alternative approach to the alignment techniques described by Weber *et al.* (1994). The major drawback of this approach is that it does not always give the correct results. To address this issue they implemented a spot editing tool into the system (Carol) where errors can be corrected manually. Clearly this is not an ideal solution.

Appel (1997), Lemkin (1997), Tekalp (2000) and Garrels *et al.* (1984) all suggested that comparison of spot lists matching are greatly enhanced if one compares the spot lists region by region, rather than spot by spot. In their current systems, this is implemented as a graph algorithm where vertex clusters are matched. In an alternative approach, described by Smilansky (2001), geometric signature of several spots that constitutes a rectangular region in a raw image, rather like a fixed group of stars has been used. Direct registration techniques to the subimage defined by this rectangular region are used which determined the correct transformation at the centre of the region. However, all these methods are based on

geometric computation and image processing techniques and they match similar protein spots rather than matching similar proteins. For example, the matching is based on the size and intensity values of the protein spots.

To detect similar spot in the target image means the identification of identical or similar proteins from the target gel electrophoresis images. In this case, the electrophoretic mobility of the proteins need to be considered to limit the search and to optimise the output. Although the spot for one particular protein in the source and target image can be identical or similar, but the following parameters can still vary:

- background value
- protein spot intensity
- protein spot shape, and
- noise in the image

For example (Figure 4.4) *triosophosphate isomerase* protein spot lies on the same line of path (*pq*) in both images *(a,b)* but the spot location and the intensity of the spot vary significantly in this case.

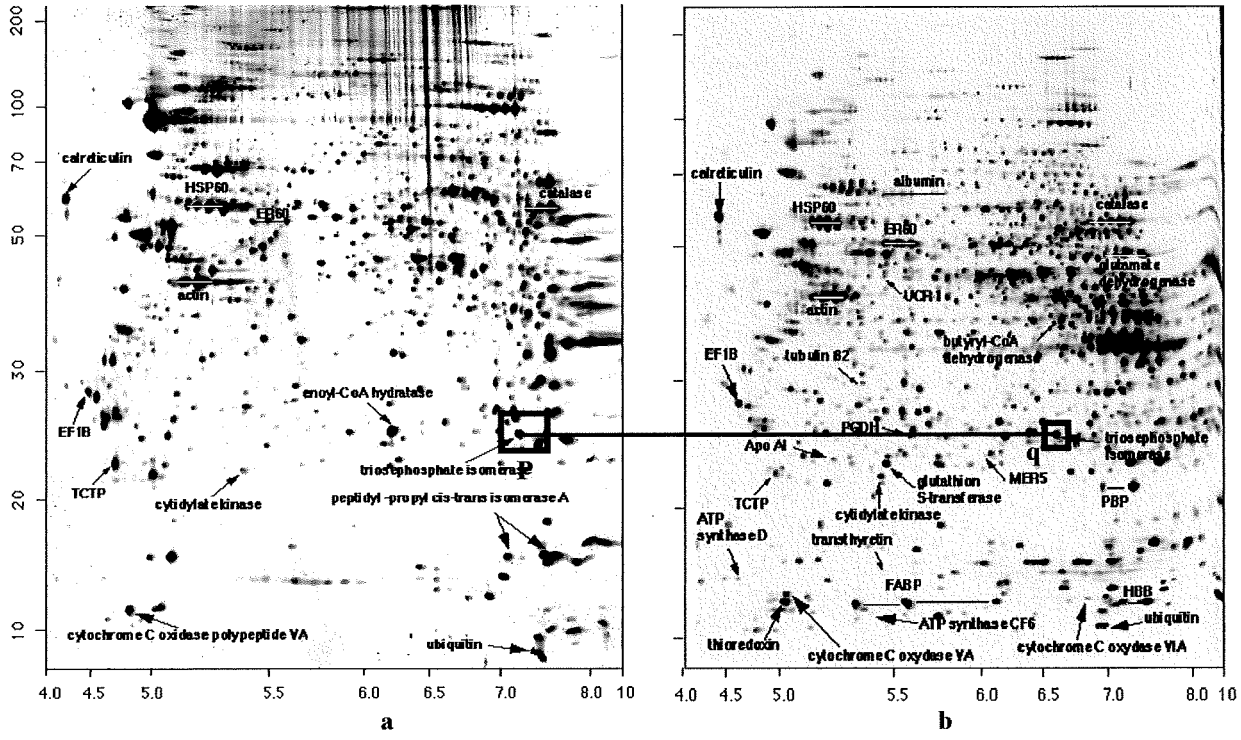


Figure 4.4. *Triosophosphate isomerase* protein spots in two different images

None of the algorithms and approaches described in the previous section (Section 4.3) have considered or taken into account the electrophoretic mobility concept to compare the gel spots. Next chapter presents a novel approach for identifying the identical or similar protein spot in 2D gel electrophoresis images. This approach considers the following factors:

- 2D gel electrophoresis protein spots differ significantly in two different images even when they represent the same protein.
- same or similar protein spots will lie on the same line of path because of their electrophoretic mobility and molecular weight.
- the intensity of the matched regions in both images can be different even though it shows a correct matching.
- the region of similar spot at the target image must lie in the same or different directional vector on the line of path.

Although many software, *i.e.* CAROL (CAROL, 1998) and MELANIE are able to identify most of the spots in an image, but a single spot identification on any line of path requires a different approach because it does not follow any conventional method of matching. To achieve this a novel approach is presented in the next section.

4.3 Spot Identification on Line of Path

A spot on line of path in gel electrophoresis can be identified in two ways:

- Point operation - for increasing visibility of the image and setting an offset value to reduce the error limit.
- Interpolation of the spots - for creating a vector of image spot intensity

Section 4.3.1 highlights the histogram and the interpolation technique for point operation. It is shown that conventional technique of point operation will not be able to identify protein spot on line of path. Hence, the new technique proposes to create a vector of spot intensity and then to identify the spots on line of path.

However, it is shown in Section 4.2 that similar protein spot may not still be present at the same positional vector as it is in the source image. In such case, it is necessary to search at the neighbourhood area. Section 4.3.2 explores the methods for searching the protein spot in the neighbourhood area.

The searching process is extended further where a region of interest from the source image is defined and it is matched with the corresponding shape of the spot in the target

image. Section 4.3.3 investigates different types of edge detectors and establishes the relevant edge detector which is most suitable for this approach.

4.3.1 Point operation on image

One of the basic approach for point operation is histogram operation. A histogram H_f of any digital image f is a plot or graph of the frequency of occurrence of each grey level in f. H_f is a one dimensional function with domain $\{0,\ldots,K\text{-}1\}$ and the possible range extends from 0 to the number of pixels in the image. Therefore counting grey levels is required to compute the image histogram and this is achieved by scanning the image. The histogram H_f contains no spatial information about f. Histogram provides grey level distribution. The grey levels with lower numbers indicates the darker pixels and the grey levels with higher numbers indicates the brighter pixels (Gonzalez, 1992).

A point operation on a digital image is a function h of a single variable applied identically to every pixel in the image. Spatial information is not available in point operation, hence, it does not affect the spatial relationships between pixels in the output image. Thus point operations do not affect neither the spatial positions of objects nor their shapes.

Linear point operation provides a grey-level additive offset L which has implications in image saturation conditions. The saturation can be resulted from the underflow or overflow condition and it leads to errors. This situation can be corrected by adding offset value L which will allow the output image $g(n)$ to set at the higher or lower end point, *i.e.* set $g(n_0) < 0$ at some coordinate n_0 to $g(n_0)=0$ and set $g(n_0)>k\text{-}1$ to $g(n_0)=K\text{-}1$. The range of offset values lies within the range of $\{0,\ldots,K\text{-}1\}$. So, the offset value L determines the shift of the histogram by the amount L to the right or to the left in order to increase the visibility of the image (Tekalp, 2000).

H_f only describes the frequency of the grey levels in f. So, to determine a vector of intensity values of the spots in gel electrophoresis image, it is required to map the coordinates of the spots. This is implemented by deriving interpolation of the images which is described in the next section.

4.3.1.1 Interpolation technique for mapping coordinates

Spatial mapping of the coordinates of an original image f to define new image g is defined as:

$$g(n)=f(n')=f[a(n)]$$

Geometric image operations are defined as functions of positions rather than intensity. The two-dimensional, two valued mapping function $a(n)=[a1(n1,n2), a2(n1,n2)]$ is usually defined to be continuous and smoothly changing, but the coordinates $a(n)$ are not generally integers. The second stage determines the values of f to define g(n) to fit the mapping in standard discrete lattice. Hence, it is required to interpolate the noninteger coordinates $a_1(n_1, n_2)$ and $a_2(n_1,n_2)$ to integer values so that g can be expressed in a standard array of coordinates.

In Nearest-neighbour interpolation technique, the coordinates are rounded prior to assigning in the matrix of g. Thus, the coordinates of the input images are mapped to the nearest integer value. In this technique the amplitude of an output image pixel to be set to the amplitude of the input pixel.

The nearest neighbour interpolation can result in a spatial offset error by as much as $1/\sqrt{2}$ pixel units (Pratt, 2001). In gel electrophoresis images, nearest neighbour interpolation can deliver inaccurate results. Because, the nearest neighbour interpolation technique rounds off many coordinates with the same value, so, it may give an impression of 'pixel blocking' or 'distorted spot edge'. A distorted spot edge can give wrong information of the protein details. This interpolation technique may also lead to a sudden change in the intensity value. Any sudden intensity change in gel electrophoresis image can lead to a new spot or can mask a spot.

Next section has proposed a new approach to determine the spot coordinates by using grey value distribution interpolation instead of using the mapping coordinate interpolation.

4.3.1.2 Determining pixel value along the cross-sections of line

The reasons for proposing an alternative approach for identifying the spots of protein in the gel electrophoresis images are as follows:

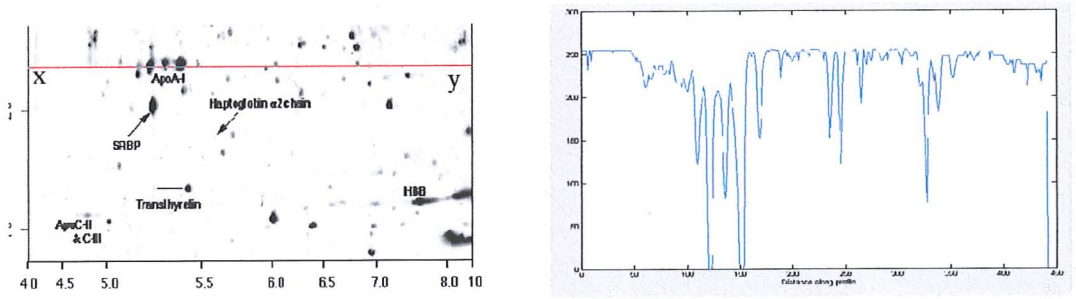
- the matching of the spots in two different images needs to be done on the basis of their respective spatial location.

- the resemblance of shape of two different spots in gel electrophoresis images does not ensure the protein similarity.
- changes in intensity values in two different images can significantly decrease the possibility of identifying any spot.

The research proposes an approach which combines the point operation technique with geometric image operation. The technique uses interpolation to find the intensity values for each point on cross-section along the path. It determines a number of point n by examining the intensity value grey image and their coordinates which touch the line of path.

A line of path is drawn (Chapter 5) on the image and it finds the approximate number of pixels through which the path traverses. It then calculates the intensity interpolation for each pixel and stores the interpolated values in a vector of *n by 3*. A two dimensional plot of the intensity values against the distance along the line segment determines the coordinates which correspond to the intensity. The peak towards the lower intensity values determines the darker region in the image. The spots of interest along the path of the image are determined by assigning a threshold value T. The choice of the threshold value T determines the trade-off between capturing all the spots along the path in the image or minimising the noise. Higher value of T can significantly reduce the capability of reducing noise.

Figure 4.5a shows gel electrophoresis image. A line of path *(x,y)* is drawn on the image. The lower pixel values which intersect the line of path are the spots on the line of path. Nearest neighbourhood interpolation technique is used to find the intensity value for each point along the line of path. It is shown (Figure 4.5b) that the protein spots which lie on the line of path show a low intensity value (peak). Hence, the lower intensity values of the pixels are chosen as the protein spots which lie on the line of path. An appropriate threshold value for low intensity is used to determine these spots on the line of path.



a b

Figure 4.5: Identifying a spot along the line of path using the low intensity values

The research has established (Section 4.2) that although a similar or identical protein spot can lie at different directional vector but it must reside on the line of path. This follows that if no spot is found on the same directional vector as that found in the source gel image, then it is necessary to identify any protein spot in the neighbourhood area on the line of path. Next section explains the approach used to identify any spot in the neighbourhood area.

4.3.2. Searching for the protein spot in the neighbourhood area

Nearest-neighbourhood search is important for matching protein spot. In any given situation, it is necessary to preprocess n objects in a metric space to determine which of the *n* preprocessed objects has the shortest distance from the query object (Brab and Knauer, 2001). In an image retrieval system, a nearest neighbour search in a metric is carried out by measuring the image similarity (for example, QBIC (QBIC, 2002)). Brab and Knauer (2001) proposed Hausdorff distance measure for nearest neighbourhood search. The outline of the Hausdorff distance measure is described in the next Section, 4.3.2.1. Hausdorff distance measure is the basis for this proposed approach (see Chapter 5). It determines the nearest neighbourhood spot in the gel electrophoresis image.

4.3.2.1 Hausdorff distance algorithm for geometric shapes

Felix Hausdorff devised a metric function between subsets of a metric space. By definition, two sets are within Hausdorff distance *r* from each other if and only if any point of one set is within distance *r* from some point of the other set (Alt, 2001 and 2000).

More formally, let there be a space X with a metric function d. For every $A \subset X$, and $r > 0$, Barnsley (1988) and Edgar (1995) defined an open neighbourhood as

$Nr(A) = \{y: d(x,y) < r \text{ for some } x \in A\}$

The Hausdorff metric *h(A,B)* is defined in terms of the neighbourhoods. For any two subsets *A* and *B* of *X*,

$h(A,B) = inf\{r: A \subset Nr(B) \text{ and } B \subset Nr(A)\}$,

where *inf(R)* is the Greatest Lower Bound of a set *R*.

Huttenlocher *et al.* (1993) defined the Hausdorff distance as a distance of compact subsets of a space if a metric measuring the distance of the points of the space already exists. More formally, they defined the Hausdorff distance as follows:

Let *P* and *Q* be compact sets in *R*. Then *H(P,Q)* denotes the Hausdorff distance between *P* and *Q*, defined as

$$H(P,Q):=\max(h(P,Q), h(Q,P)), \text{ with}$$

$$H(P,Q):= \sup_{x\in P} \inf_{y\in Q} \|x-y\|$$

Brab and Knauer (2001) outlined an algorithm based on Haudorff's distance theory to perform the nearest-neighbour queries among point sets. They described the algorithm in a situation where a given data set $R = \{Q1,\ldots,Qk\}$ that consists of k point sets $Q1,\ldots,Qk$ in space *D*. Given a query point set *P* in *D* to determine:

$h(P,R):=\min_{Q\in R} h(P,Q)$, the smallest directed Hausdorff distance that *P* has to an image in *R*

$h(R,P):= \min_{Q\in R} h(Q,P)$, the smallest directed Hausdorff distance that an image in *R* has to *P*

Huttenlocher *et al.*, (1993), Alt *et al.*, (1995, 2000 and 2001) noted that the Hausdorff distance is very sensitive to even a single point. For example, consider, where the point *x* is some large distance *D* from any point of *A*. In this case *H(A,B)=D* which is determined solely by the point *x*. Thus rather than using *H(A,B)* a generalization of the Hausdorff distance (which does not obey the metric properties on *A* and *B*, but does obey them on specific subsets of *A* and *B*) is used. This generalized Hausdorff measure is given by taking the k-th ranked distance rather than using the maximum, or the largest ranked one,

$$h_k(P,Q):= \text{kth}_{x\in P} \min_{y\in Q} \|x-y\|$$

where *kth* denotes the *k-th* ranked value (or equivalently the quantile of m values). For example, when *k=m* then *kth* is *max*, and this is the same measure as *h(P.Q)*.

The generalised Hausdorff distance is also used by Alt *et al.* (2001) for matching point patterns and triangles in three dimensions.

It can be concluded that the Hausdorff distance measures the extent to which each point of a model set lies near some point of an image set or vice versa. This distance can then be used to determine the degree of resemblance between two objects that are superimposed on one another (Huttenlocher *et al.* 1993 and Ehrenmann *et al.* 2000). However, the Hausdorff distance measure approach can not be applied directly to the gel electrophoresis image protein spots because any single protein spot model in image *A* can identify a set of spots in image *B* which are similar protein spot. This approach is not very helpful to extract only those spots which have the capability to be the identical or similar protein spot. Panek and Vohradsky, (1999) used another approach to detect neighbourhood spots. They argued that each electrophoretic spot can be assigned a neighbourhood consisting of a defined number of closest spots. They concluded that due to the same electrophoretic conditions, point patterns of matching neighbourhood spots in the reference and to that in the compared gels will have more similarities than the point patterns of other spot pairs. They defined a neighbourhood descriptor and the matching is then reduced to a comparison of the descriptor. In this technique, the surrounding of each spot is divided into half-plane for segmentation purpose and descriptor for all the spots are stored. This technique is solely based on image registration and it concentrated on matching the similar spots instead of matching the similar protein spots. Panek and Vohradsky (1999) also noted that the drawback of this approach is that any spots close to the borders of the segment in the reference pattern can appear in different segments in the matched pattern which will obtain a different descriptor. This invalidates its correct interpretation of the reliability of the whole pattern description and comparison. Furthermore, this process has the risk of matching more than one candidate spots if it contains identical neighbourhood descriptor.

The objective of this research is to identify identical or similar protein spot in the target gel electrophoresis images. It uses electrophoretic mobility feature of the protein. The next chapter (Section 5.2.3) explains the approach which has been used to detect the neighbourhood protein spot. The basis for this alternative approach to match similar protein spot is that the similar or identical protein spots will lie either on the same line of path or on the inter-section of the line (plane divider). It will not lie in any half-plane (plane that is on the either side of the line of path) of the image.

the presence of White Gaussian noise (Pratt, 2001). It also includes the elements of the Laplacian approach.

Canny operator detects the edges by convolving a one dimensional continuous domain noisy edge signal *f(x)* with an antisymmetric impulse response function *h(x)*, which has zero amplitude outside the range *[-W,W]*. An edge is marked at the local maximum of the convolved gradient.

The method has three distinctive goals:

Good detection: the amplitude signal to noise ratio of the gradient is maximised to obtain a low probability of failure to mark real edge points and a low probability of falsely marking nonedge points. Canny assumed that false-positive and false-negative detection errors are equally undesirable and so gave them equal weight.

Good localisation: edge points marked by the operator should be as close to the centre of the edge as possible. Canny further assumed that each edge has nearly constant cross section and orientation, but his general method includes a way to effectively deal with the cases of curved edges and corners.

Single response: there should be only a single response to a true edge. The distance between peaks of the gradient when only noise is present, denoted as x_m, is set to some fraction k of the operator width factor W. Thus

$$X_m=kW$$

Canny (1986) proposed unifying the set of edge maps into a single result by using a technique called 'feature synthesis' which proceeds while tracking the edge segments. Canny's (1986) method includes a 'goodness of fit' test to determine if the selected filter is appropriate before it is applied. The test examines the gray-level variance of the strip of pixels along the smoothing direction of the filter. If the variance is small, then the edge must be close to linear, and the filter is a good choice. A large variance indicates the presence of curvature of a corner, in which case a better choice of filter would have smaller extent in the smoothing direction (Pratt, 2001).

Canny's method looks for zero crossings of the second derivative, like Laplacian approach. This is a different perspective to understand the essence of Canny's method. The Laplacian second derivative is nondirectional. Canny is performed only in the gradient direction, directly across the local edge. A derivative taken along an edge is counter

There are several algorithms reported by different researchers to detect the contour of the spots in the gel electrophoresis images. The algorithms ranges from Gaussian fitting to Laplacian of Gaussian edge detectors (Appel *et al.*, 1997; Anderson *et al.* 1981; Lemkin *et al.*, 1981 Prehm *et al.*, 1987 and Bettens *et al.*, 1997).

Marr and Hildreth (1980) reported on Laplacian of Gaussian (LoG) edge detector (Marr and Hildreth operator) based on filtering the image with a Gaussian Kernel selected for a particular edge scale. In this approach Gaussian smoothing operation limits the image to a small range of frequencies and reduces the noise sensitivity problem when it detects the zero crossing. This creates a set of edge maps as a function of edge scale. Each edge point is considered to be resided in a region of scale space. This scale space can then be used to analyse the edge maps.

A significant advantage of using LoG is that the Gaussian function is smooth and localised in both the spatial and frequency domains. This provides a good compromise between avoiding false edges and minimising errors in edge position. However, implementing the LoG, requires to construct a filter and to use the filter to convolve with the image. It also requires to construct the filter large enough to avoid significant truncation effects. Construction of appropriate filter is a manual process and it utilises significant amount of processing time and memory space.

Pleibner *et al.* (1999), used Watershed Transformation to determine the optimal contour of protein spot. In this technique, a monochrome image is considered to be an altitude surface in which high-amplitude pixels correspond to ridge points, and low-amplitude pixels correspond to valley points. The assumption is that valleys of the gradient image correspond to the requested regions whereas the ridges-watershed define the optimal contour of a region. Each region is then described by a feature, achieving a so-called mosaic image. This approach tends to produce results that are oversegmented because of the image noise amplification calculated by the gradient image (Lopez, 1999). Moreover, Gaussian smoothing needs to be performed as a prefilter.

Most of the differential edge enhancement operators, *e.g.*, Roberts operator (Roberts, 1965), Prewitt's edge gradient operator (Prewitt, 1970), Sobel's operator (Gonzalez, 1992) *etc.* are derived heuristically. However, Canny (1986) developed an analytical approach to the design of gradient operators based on an one dimensional continuous domain model in

4.3.3 Point pattern matching in gel electrophoresis images

In biomedical technology it is often required to compare a whole series of 2D gel electrophoresis images to understand the underlying disease mechanism and to elucidate the protein expression. However, the following difficulties exist for such automated comparisons of these images.

- inaccuracies in the electrophoresis process
- different format of electrophoresis images
- different spot expression
- different resolution of the images

Therefore, it is necessary to investigate the spots in the target image and to look for the resemblance of the spot patterns both geometrically and spatially for automation. Even finding the partial resemblance of the spot patterns between source and target image can contribute to a new era in molecular biology.

Spot detection itself is a complex process. A list of spots of interests is determined by the approach described in the previous Section. Next chapter describes the approach used to detect the protein spot which is identical or similar to the source spot found within this list. The following sections (4.3.3.1 and 4.3.3.2) examine the appropriate edge detector which will be used to determine the contour of the spots in source image and the result will then be used to compare with the protein spot shape in the target image.

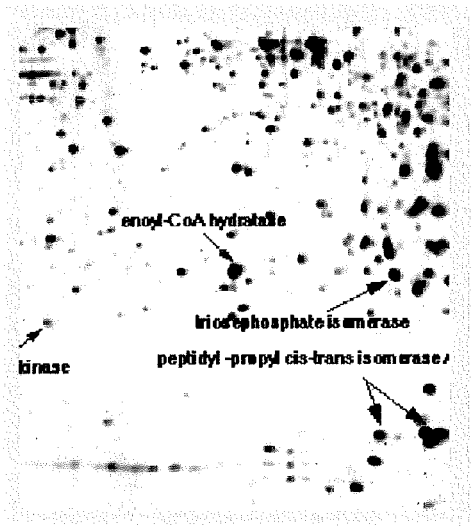
4.3.3.1 Edge Detectors

Edge detection is the process of determining which pixels are the edge pixels in an image. The result of the edge detection process is typically an edge map. A derivative or gradient is the basis for an edge detector. To illustrate the idea, there must be a point x_0 that marks the transition from the low-amplitude region to adjacent high-amplitude region. The gradient approach to detect this edge is to locate x_0 where gray level function $f(x)$ reaches a local maximum or, equivalently, the function reaches a local extremum. In Laplacian approach, x_0 is a point where a function $f'(x)$ crosses zero.

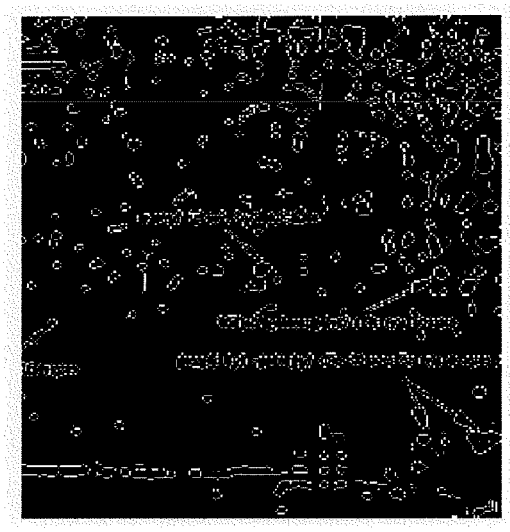
To deal with 2D images the Laplician or gradient approach needs to be extended to deal with discrete cases. The discrete nature of digital images requires approximation of derivatives to avoid noises, crosstalk or interference resulted from the nearby edges.

productive because it introduces noise without improving edge detection capability. By being selective about the direction in which its derivatives are evaluated, Canny's approach avoids this source of noise and tends to produce better results.

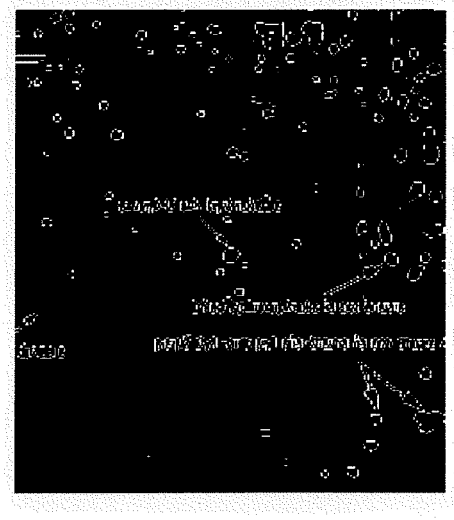
A comparison of different edge detectors on gel electrophoresis images are presented in Figure 4.6. It is notable that Canny edge detector identifies more edges than any other edge detectors. It is also evident from the pictures that Sobel and Prewitt operators show more broken edges than Canny edge detector.



Original (a)



Canny Operator (b)



Sobel Operator (c)

Prewitt operator (d)

Figure 4.6. Detection of spot edges using different edge detectors, (a) original image, (b) Canny edge detector, (c) Sobel edge detector and (d) Prewitt edge detector.

4.3.3.2 Edge linking method: Hough Transform

Edge lining techniques are very useful methods for joining edges or creating a boundary of a certain area. For protein spot pattern matching it is required to identify the shape of the source spot so that the shape can be analysed with the target spot shape for finding resemblance.

The Hough Transform (HT) can be used as a mean of edge linking (Hough, 1962). It involves the transformation of a line in Cartesian coordinate space to a point in polar coordinate space. A straight line in Hough transform described parameterically as distance of the line from the origin (ρ) and an angle (θ) of the origin with respect to the x axis. So, the line can be described parametrically as:

$$\rho = x \cos \theta + y \sin \theta$$

The Hough transform of the line is a point at coordinate (ρ, θ) in the polar domain. A family of lines passing through a common point maps into the connected set of ρ-θ points.

The feature space of Hough Transform is required to be quantised to certain parameter states after the minimal and maximal values for each parameter are determined. Each flow vector votes for a set of quantised parameters. The parameter sets that receive at least a predetermined amount of votes are likely to represent candidate edges. The drawbacks of HT are its need for significant amount of computation and memory. To overcome these problems modifications of HT have been proposed by Tekalp (2000):

- decomposition of the parameters space into two disjoint subsets to perform two 3-D Hough Transforms
- a multiresolution Hough transform, in that at each resolution level the parameter space is quantised around the estimates obtained at the previous level
- multipass Hough technique, in which the flow vectors that are most consistent with the candidate parameters are grouped first. In the second stage those components formed in the first stage that are consistent with the same flow model in the least-square sense are merged together to form segments.

Duda and Hart (1972) used Hough transform technique for line and curve detection in discrete binary images. Each nonzero data point in the image domain is transformed to a curve in the ρ-θ domain which is then quantised into cells. If an element of a curve falls in a cell, that particular cell is incremented by one count. Once all points are transformed, the ρ-θ

electrophoresis images with the structural protein image is used. Section 4.4.1 highlights this approach and shows how this meta-data database, developed by Lemkin (1997), can act as a mediator to link with the 3D structure of the protein.

4.4.1 From gel meta data to 3D structural protein image

The 2DWG meta database provides details of gel electrophoresis images. It also provides access to other servers such as Expasy and swiss-prot (URL: http:\\www.expasy.org/ch2d/). These two servers act as gateway to provide the details of any protein. By closely examining the expasy and swiss-prot servers it is possible to correlate gel spot with the swis-prot entries for protein biological details and then with PDB entries for protein structural details. When any image is selected from the 2DWG database (Figure. 4.8), it provides access to the expasy server for further details. Expasy server provides the list of 2D page images maintained by the Swiss-prot. All the spots in the images are marked. For example a list of gel electrophoresis images are provided in Swiss-prot database (Figure 4.8). When any gel image is loaded, one can click on the marked spot for further details. For example, by clicking on spot (p) in the gel electrophoresis image (Figure 4.9a), the details of the spot can be retrieved (Figure 4.9b). The swiss-prot details of the spot provide the information about the entry name, accession number, gene name and references. This page also provides the information about the EMBL (URL: http://www.ebi.ac.uk) entries and PDB entries (Figure 4.9c). The PDB entry leads directly to access the PDB structural page for further viewing of 3D structural details of the protein (Figure 4.9d). However, all these processes are manually carried out, for example, selecting the protein spot to find out the protein accession number, accessing the SWISS-prot database and manually matching the accession number for 3D structure. This process is also independent of the preceding transactions. The target (3D structural protein image) image can be reached by clicking the relevant links provided by the web pages. But the process does not allow to compare the spots of the local database with the image gel spots available in the public domain databases and it does not retrieve the images dynamically, *i.e.*, without clicking at all the different links. 3D structural protein image retrieval is necessary for dynamic spot analysis to make conclusion about the local gel image spot. It also helps to make clinical decisions by comparing both the 3D structural protein images of the spot of local gel and that of the global gel image. Therefore, it is necessary not to restrict the comparison within the image spots,

here to create an additional relation between protein spot and the 3D structure of the protein to establish the conclusive evidence of matching similar or identical protein. By correlating protein spot with the 3D structure of the protein can not only aid the comparison of phenotypic details of the diseases but it can also lead to the drug discovery process. However, these approaches have not yet been reported by the researchers.

Lemkin (1997) developed an internet based meta-data database of 2D gel electrophoresis images (2DWG). The meta-data database includes source of proteins, species, image location, location of the database containing protein details and spot map. This meta-data database can be searched by using the conventional search engine but the searching is based on texts. A search result is presented in Figure 4.7. Searching is carried out either by web gel ID for example, WG00123 or by using species and protein source, for example, human and liver respectively. Once the search is completed, the raw image can then be downloaded from the resource and the other details are available in row and column format.

List of 2D gel images

Search expression: human AND Liver AND map

Tissue or Organelle	Species	Cell line/ Strain	Image URL	DB URL	Isotope stain Ab	CA/ IPG	IEF/ NEPH. GE	pH range	Mr (Kd) range	Lab/ Org/ Comp	Scan/ synth/ diagrm	Map/ raw data	Miscellaneous	2DWG ID #
Liver	Human	-	Image	DB	Silver	IPG	IEF	3.5-10	5-250	ExPASy	scan	map	-	WG00079
Liver	Human	-	Image	DB	Silver	IPG	IEF	3.5-10	5-250	ExPASy	scan	map	Fig 1, 1993	WG00080
Liver	Human	-	Image	DB	Silver	IPG	IEF	acid	5-250	ExPASy	scan	map	Fig 2, 1993	WG00081
Liver	Human	-	Image	DB	Silver	IPG	IEF	basic	5-250	ExPASy	scan	map	Fig 3, 1993	WG00082
Liver	Human	-	Image	DB	Coom	-	IEF	4-8	-	LSB	scan	raw	LSB std human liver	WG00088
Plasma	Human	-	Image	DB	Coom	CA	IEF	4-8	10-150	IUPUI/ Mol Anat Lab	synth	map	Human liver like toxicology	WG00229
Liver	Human	-	Image	DB	-	IPG	IEF	4-6.5	5-250	ExPASy	scan	map	Fig 2(A) (Electrophoresis) (1995) 16, 1131-1151	WG00263
Liver	Human	-	Image	DB	-	IPG	IEF	5.5-10	5-250	ExPASy	scan	map	Fig 2(B) (Electrophoresis) (1995) 16, 1131-1151	WG00264

There were 8 entries found matching the query.

Figure 4.7. Result of search executed on 2DWG meta-data database.

The main objective of creating this meta-database is to keep track of the gel electrophresis images. As an additional benefit, it also provides web indexing search engines for the URL list of the gel electrophoresis images. The 2DWG provides a tissue specific lists of gel electrophoresis images regardless of which laboratory protocols were used to create these images. However, this meta-data database does not add functionality for comparing the image spots. Instead it relies on visual comparison with the local gel images to identify desired protein spot. Furthermore, it does not also provide any link to 3D structural protein which corresponds to the protein spot. This research investigates this meta-data database of gel electrophoresis images as an intermediate mediator to link local gel electrophoresis image with the 3D structure of the proteins. The same concept of meta-data based linking of gel

cells are examined. Large cell counts correspond to colinear data points that may be fitted by a straight line with the appropriate ρ-θ parameters. If we consider a line segment in a binary image $F(j, k)$ which contains a point at coordinate (j,k) that is at an angle ϕ with respect to the horizontal reference axis, then when the line segment is projected, it intersects a normal line of length ρ originating from the origin at an angle θ with respect to the horizontal axis. The Hough array $H(m,n)$ consists of cells of the quantised variable ρ_m and θ_n.

The approach as described above can be used for edge linking to establish the spot shape in the source gel electrophoresis image. Ballard (1981) proposed Generalised Hough Transform technique to determine any shape. In his technique a contour model is approximated by a set of sample points M. Every point in M can be described with respect to some reference point y inside the contour through the vector $r_i=y-x_i$. The r's are stored in a R-table. For the edge image I', an array H of same size of image I' is created. The edge image can be obtained by Canny edge detection technique. The Generalised Hough Transform maps every edge pixel P_E in I' to a set of points P_H in H, called Hough buffer. Every element in H is incremented by one if it falls in P_H. This method can easily be enhanced for curvature or rotated contours by introducing new Hough accumulator for every angle ϕ and for rotation of the contour δ. The R table is enhanced accordingly.

In Hough Transform, any cell whose magnitude is sufficiently large defines a straight line. If this line is overlaid with the image edge map it should fill the missing links of straight line edge segments, and therefore, it can be used as a mask to fill in the missing links. These straight lines are used as region of interest (ROI) mask that controls the edge as such that the processing is performed only on edge map pixels with the ROI. Edge map pixels outside the ROI are left unchanged. Next chapter shows how the above process is applied to create a ROI in the source gel electrophoresis image and the ROI shape is then compared with the target image ROI shape for shape variance analysis.

4.4 Dynamic Linking of Protein Spot to 3D Structure

Gel electrophoresis images play a key role in biomedicine engineering ranging from clinical purpose to the drug discovery process. The key purpose of analysing these images is to match the protein spots for identical or similar protein. It has been established in the previous sections (Section 4.3) that the spot matching does not necessarily give any conclusive evidence of identifying identical or similar proteins. The research is proposing

instead it is essential to broaden the search from the gel spots to 3D images of protein which can then be compared for different clinical and drug discovery purpose.

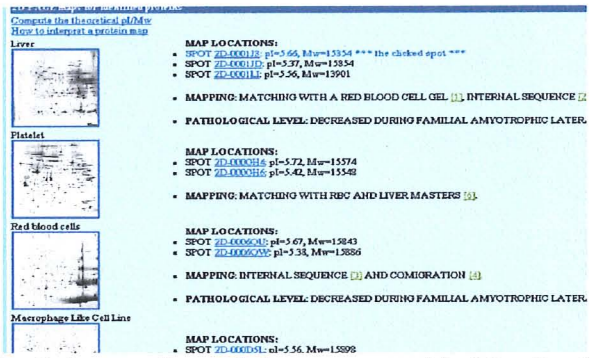


Figure 4.8. A list of gel electrophoresis images provided by the Swiss-prot database.

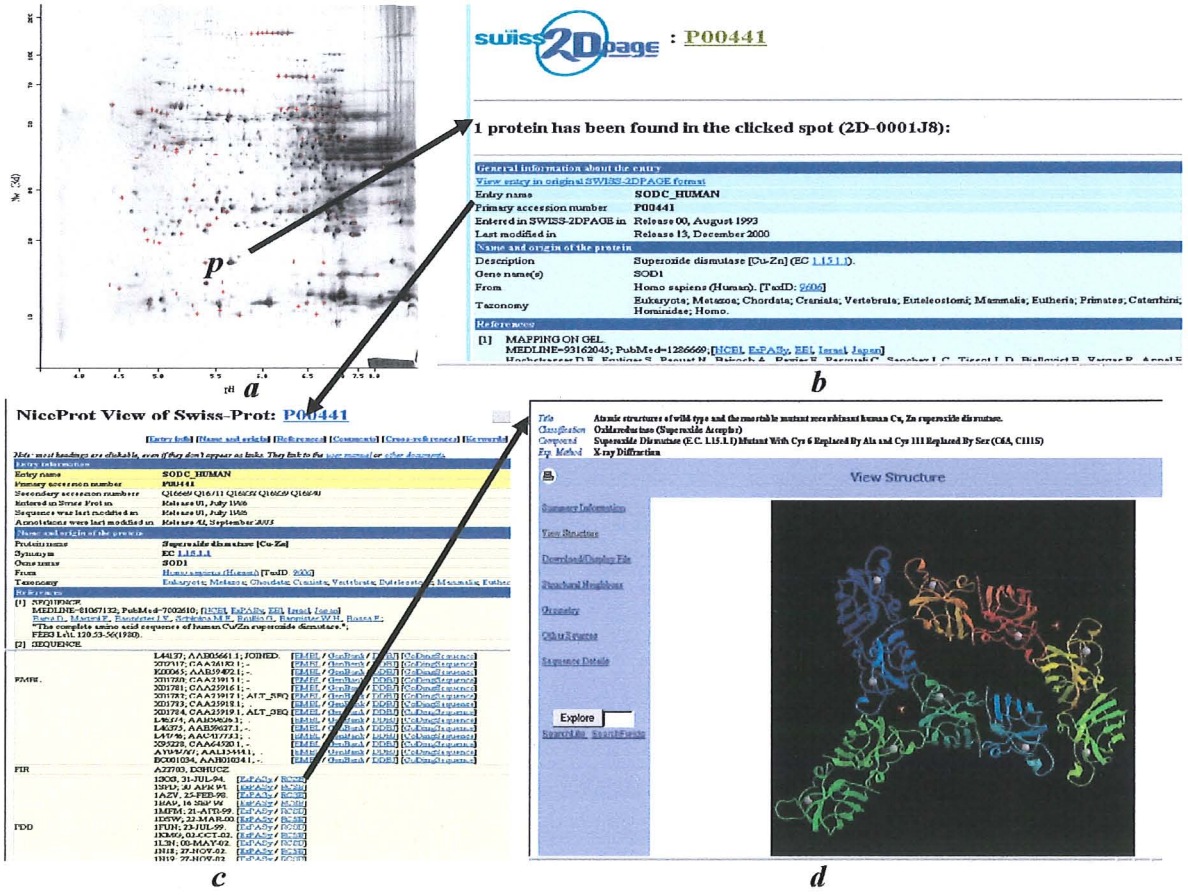


Figure 4.9. (a) A protein spot is selected at position p, (b) the spot details are retrieved from swiss-2D database, (c) further details are retrieved from Swiss-prot and then (d) the 3D structural image is retrieved from PDB.

Present methodologies do not use such dynamic linking. This research thus proposes an alternative approach to link spot dynamically with the 3D structure. Next chapter (Section

5.4) describes how to link the local gel image with the global gel image for spot comparison. It also shows how the 3D structural image can be linked with the image spots for further comparison.

4.5 Summary

In this chapter the laboratory technique for generating gel electrohoresis images is described. A corodinate for any protein spot in the 2D gel image is determined by its electrophoretic mobility, *i.e.* by placing its isoelectric charge in X axis and its molecular weight in Y axis. Hence all the spots with same molecular weight (same Y values) will lie on a specific line of path. The spots with the same Y values are considered as spots of interest for matching the source image protein spot with the target protein spot.

A number of software and algorithms are presently available to identify and to match the protein spots in different images, namely, Melanie (Appel *et al.*, 1997), CAROL (http://www.gelmatching.inf.fu-berlin.de) and Flicker (Lemkin, 1997). These software only works well when the standard image environments are maintained, *e.g.*, landmark setting by the user, same intensity value, same concentration of protein, and same lane. These software also require correcting errors manually. These software also do not take into consideration the electrophoretic mobility of the protein. These software match the selected spots of the protein in any location of the image and which does not necessarily match the required proteins corresponding to the spots. The research shows how to identify the protein spots on the line of path as these protein spots are considered as same or similar protein corresponding to the source image protein spot. Different methodologies are explored to achieve this. Point operation on image, *i.e.* histogram or interpolation, does not provide the desired objective to create an array of spot location. An approach is then outlined to derive an array of spot locations on any line of path by interpolating the intensity values on the cross-section along the path. A plot is created out of this array. The plot with peaks within a threshold value is considered as a point with spots.

In some cases, the spots may not exist in a location as it is present in the source image. In such case, it is required to identify the spots in the neighbourhood area. The major challenge of searching any spot in the neighbourhood area is to locate the spot at the closest position of the original positional vector. Housdroff Distance methodology is explored for locating the spot in nearest neighbourhood area. The algorithm measures the minimum directed Housdroff

distance between two points in a dimension D. It is shown that this approach can not be used directly for neighbourhood search as it can only search for all the similar spots in the surrounding area but it can not search for the spot which lies on the line of path. Next chapter suggests an alternative approach for locating the neighbourhood spot by utilising the nearest directional vector concept using the distance and the angle information. It extends the approach further by analysing the shapes of the spots.

Various approaches used for edge detection to determine the shape of the target image are also explored. It is found that the Canny edge detector gives better results in comparison to Sobel and Prewitt edge operator. Canny detector can identify more shapes and it has the capability of correcting errors. It is concluded that the Canny edge detector can be used for shape detection in the target image. However, to compare the shape with the target image, methodologies used for detecting the shape of the source image are analysed. Hough Transform method is used to link the edges if any set of points are given in a region of interest masking. Next chapter describes the method for determining the source spots shape by defining a region of interest.

Many algorithms identify protein spots in gel images and they determine their shapes by using clustering approach, for example Panek and Vohradsky, (1999) and Appel *et al.* (1997). These algorithms match the protein spots based on image processing or geometric techniques, but the algorithms do not find any similar or identical protein from this searching. In addition to the image processing or geometric technique the protein spot needs to be linked with the 3D structure of the protein image to achieve this. Swiss-Prot listed a number of gel electrophoresis images and maps its spot with the 3D structure of protein. However the existing technique to retrieve gel image and its corresponding 3D structural protein is conducted manually and it requires multiple clicking on the links to reach the target image. This research proposes that the retrieval of 3D image should be automated and a dynamic link needs to be established when any spot in the gel image is selected. Next chapter proposes the alternative technique to overcome the drawbacks of the current approaches. The process is described in the next chapter and it explains how any spot in the source image is identified interactively by the user and then how the spot is matched in the target image. It also avoids selecting multiple links to reach the target 3D images of the protein. 3D image is retrieved dynamically as soon as the target gel image corresponding to the matched spot is retrieved.

Chapter 5

Detection of Identical or Similar Proteins from 2D Gel Electrophoresis Images

Chapter Objective

The approach presented in this chapter uses a novel technique based on electrophoretic mobility to match protein spots between source and target images. The algorithm identifies the protein spot in the target image which lies on the same line of path as it is in the source image. A shape matching algorithm using Generalized Hough Transform and Canny Edge Detection method is used to determine the shape variance. The method as described here achieves an accuracy of 90% or more in identifying the same or similar proteins from the target image. Finally, a dedicated target based database has been created to store a set of finite values of an element spot for correlating the 3D protein structure.

Chapter Contents

Chapter 5

Detection of Identical or Similar Proteins from 2D Gel Electrophoresis Images

5.1 Introduction

The 2D gel protein images with detail maps are available now in the public domain databases, for example, ExPASy SWISS-2PAGE (ExPASY, 2000 and SWISS-2D, 2000). However, a scalable and robust algorithm for comparing and analysing single protein spot has not yet been developed. The previous chapter presented some of the drawbacks of the present approaches for analysing protein spots (Chapter 4 Section 4.2).

At present the researchers use gel electrophoresis protein spot comparison for analysing the laboratory results with the existing clinical features. To compare the protein spots the researchers either need to compare the gels using manual visual techniques or they need to use analysing tools for automated comparison with other gel databases. It is almost impossible to compare the gel spots manually because the volume of similar spots in a single image is huge. For automated comparison, the laboratories need to be equipped locally with the gel databases and the analysing software and again it is not an easy task for comparing such protein spots.

The comparison of two gel image protein spots requires identifying the target protein spot's positional vector, its morphological description and its correlation with the target spot. To achieve this, gel comparison software needs not only to identify the spots with similar morphology but it also needs to determine the electrophoretic mobility of the target protein. The mobility of a spot can be determined by using the positional vector of the target spot and then by comparing it with the source spot. This approach helps to limit the scope of the searching region. Once the initial target spot is identified then further

analysis for morphological correlation between the source and target spot is required. However, the challenge is not only to compare but to create such a comparison algorithm which will be able to perform comparison among the inter-laboratory gel images via internet (Efrat *et al.*, 2001).

This chapter concentrates on the computational task for spot comparison and how to extend the limit of searching. The approach for spot comparison described here addresses the following issues of gel image spot matching (Khan and Rahman, 2003):

- identifying a target spot based on the electrophoretic mobility of protein spot.
- identifying the source and target spot regions to restrict the region of searching.
- correlating source and target spots based on computational geometry, morphological description and spot intensity resemblance.
- correlating dynamically the target spot with the 3D structural protein for more details about the spot.

The initial tasks of the proposed approach are to determine the positional vector of the source spot and to determine the region of interest. This is described in Section 5.2.1 and 5.2.2. Section 5.2.3 explains how a set of parameters are derived from the region of interest and then how these parameters are compared with the target spot's region of interest. Section 5.2.3.1 describes the approach for nearest neighbourhood spot search. The algorithm looks for nearest neighbourhood spot on the line of path (Chapter 4, Section 4.1) when no spot is found on the same positional vector as it is in the source image. In the case of nearest neighbourhood search the algorithm also determines the positional variances with the source protein (Section 5.2.3.2 (i)). The algorithm also determines the morphological variances. Generalised Hough Transform technique and Canny edge detection method are combined to determine the shape resemblance. This approach is described in section 5.2.3.2 (ii). Section 5.2.4 describes the approach for correlating the 3D structure of the protein with the target protein spot. This is carried out by extracting the image feature of the spot and then linking with the meta data. Section 5.3 presents the experimental results. Finally, section 5.4 presents a discussion and summarises the chapter.

5.2. Methodology

The method described in this section is used to determine the position of similar or identical protein spot in target image which lies on the line of path. For shape matching the method interprets the pixel regions in terms of directional vector, intensity, contour description, and its electrophoretic mobility. It identifies a set of pixels with its centre and draws a polygonal curve around these set of boundary pixels to determine the shape which establishes the region of interest. It then searches for the same spot at the same or least variance position in the target image. The shape of the spot 'ρ' needs to be matched with the shape 'λ' which can be found at the same or least variant position in the target image. To identify the identical or the similar spot in the target image the following criteria need to be fulfilled:

1. Intersecting the spots in the target image with the same line of path as it is used for the source image spot
2. Determining the spot of interest in the target image by examining the directional vector
3. Determining the shape of the source spot for resemblance matching with the target spot shape.

The objective is to achieve an 'exact match' to identify the spot in the target image which has close resemblance to the source image spot features.

5.2.1 Determining the position of the protein spot in the source image

The source image S from which a particular spot of protein will be detected is set to a user defined reference point with equal width and height. This assumption is applied to all the source images. The source image is also divided into four quadrants, two upper regions (a and b) and two lower regions (c and d) with the horizontal and the vertical axes intersecting at the centre point of these quadrants (Figure 5.1). Any protein spot 'p' of the electrophoregram can lie in any one of these quadrants.

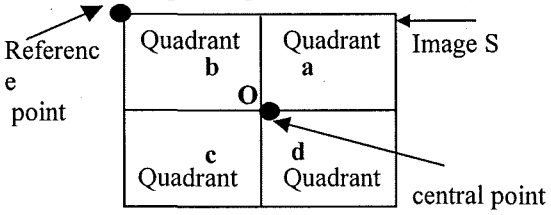


Figure 5.1. Source image divided into four quadrants

If we consider this 'p' as any point on the vertical plane, we can define a vector on this plane to determine the orientation of the spot. The vector (δ_s, θ_s) to reach point 'p' from centre of the image will produce the specific orientation of the spot (Figure 5.2), where δ_s and θ_s are the path length and the corresponding angle respectively.

The directional vector, δ_s, from the centre point to point 'p' is determined as follows:

$$|\delta_s| = \sqrt{(x_o - x_p)^2 + (y_o - y_p)^2} \quad \ldots\ldots\ldots \text{(i)}$$

where (x_o, y_o) is the coordinate of the central point O and (x_p, y_p) is the coordinate of point p. The angle with horizontal axis, θ_s, is determined as follows:

$$\theta_s = \text{arctg}\left(\frac{y_p - y_o}{x_p - x_o}\right) . \quad \ldots\ldots\ldots\ldots\ldots\ldots \ldots\text{(ii)}$$

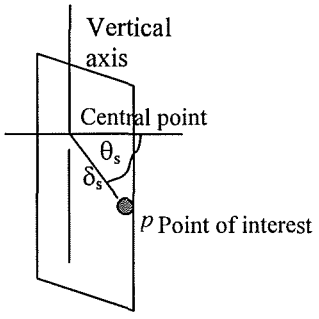


Figure 5.2. Angle produced with the horizontal axis for any point of interest on the vertical plane.

δ_s and θ_s will determine the orientation of point 'p'. A region of interest is drawn around the point to determine its mean pixel value (M_s) (Figure 5.3).

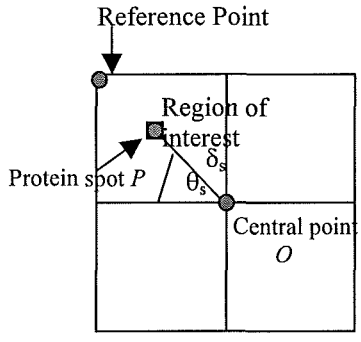
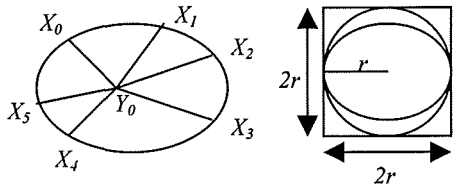


Figure 5.3. Region of interest of point *p* in the source image.

5.2.2 Defining the region of interest

The region of interest is defined by a set of boundary contour points *Xs* which are selected interactively by the user where the contour points, *M*, is defined as $M = \{X_1, X_2, ..., X_n\}$. Each contour point of *M* is defined with reference to the centre point Y_0. For each of these points, a vector *r* is defined. The largest *r* is taken as the radius and a circle with radius *r* is drawn which covers the whole spot. A rectangle of *2r* width and height is then drawn as the region of interest (Figure 5.4).



a b

Figure 5.4 (a) A set of points defined by the user, (b) Defining the region of interest.

5.2.3. Matching the selected protein spot in the target image

The target image '*T*' will be loaded from local or public domain databases at a user defined reference point. To locate the protein spot on the target gel electrophoregram at the same position and orientation, image *T* is also divided into four quadrants. The path length, δ_T and the angle with the horizontal axis, θ_T are determined for the target image using the previous equations (i and ii). Region of interest is drawn at the same position and orientation and the mean value, M_T is calculated. If the mean value M_T matches with the mean value M_s at the same position and orientation, then the point will be considered as matched point. The target point will be considered as 'matched point' when:

i. $\theta_S = \theta_T$

ii. $\delta_S = \delta_T$ and

iii. $M_S = M_T$

In other situation where

i. $\theta_S = \theta_T$

ii. $\delta_S = \delta_T$ and

iii. $M_T < M_S$ or $M_T > M_S$ (where M_T is within a threshold value),

it will be considered as 'spot found' and it might be the same or similar protein. The mean value of the region of interest may vary due to the variation in the protein

concentration or variation in the image brightness/contrast. If no spot is detected on the line of path at the same location it will then continue to search for the protein spot in the neighbourhood area.

5.2.3.1. An efficient approach for neighbourhood spot searching

The neighbourhood spot identifying method is formalised as follows:

Let P be finite set of line segment in space R. Then the nearest neighbourhood spot S is attained either at an endpoint of a line segment in P or at an intersection point of a line segment in P. When moving along a line segment e in P, for a query point m the distance is maximal at an endpoint of the line and minimal at the point n where d(m)=d(n); d(m) denotes the distance for point m and d(n) denotes the distance for point n.

The line of path through the region of interest for the neighbourhood spot matching operation is chosen on the assumption that all identical or similar protein spots will lie on this path. The assumption is based on the fact that the electrophoretic mobility and molecular weight of the same or similar protein do not vary. The line of path drawn at θ_T angle with δ_T length is parallel to the horizontal axis and it goes through the region of interest on image T. The line will intersect all similar points along its path. The line will be considered as a 'non emptied' line of path if it intersects through at least one spot of the protein along its path (Figure 5.5).

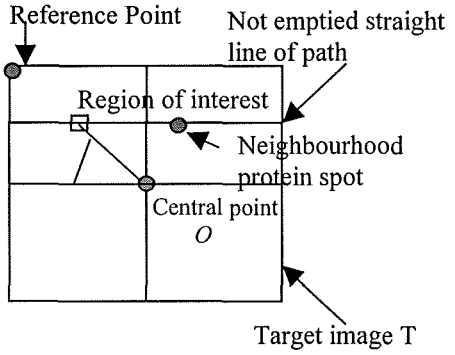


Figure 5.5. Neighbourhood protein spot - non emptied straight line of path in the target image

The linear candidates for matching are determined by sweeping along the line of path. The sweeping criteria are defined as follows:

i. If region of interest lies at quadrant 'b' then it will look in the right direction along the path (towards quadrant 'a')

ii. If region of interest lies at quadrant 'a' then it will look in the left direction along the path (towards quadrant 'b')

iii. If region of interest lies at quadrant 'c' then it will look in the right direction along the path (towards quadrant 'd')

iv. If region of interest lies at quadrant 'd' then it will look in the left direction along the path (towards quadrant 'c')

Figure 5.6 illustrates the directions of search for the neighbourhood spot.

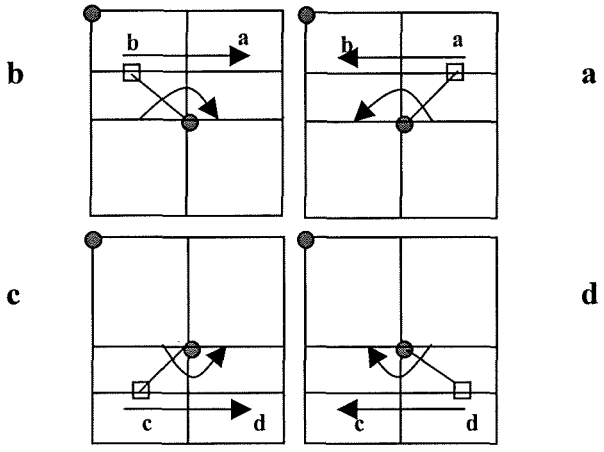


Figure 5.6. Directions of search for the neighbourhood spot.

5.2.3.2 Selecting the best matched spot

i. Directional vector variance:

The line of path is drawn to isolate all those spots from target image which lies on the line of path. A vector for all these spots are stored. The vector $\vec{v}$ consists of pixel value MT_n, length δ_T, angle θ_T and the coordinates of the spots. $\vec{v}$ can be expressed as

$$\vec{v} = \begin{bmatrix} MT1 & \delta T1 & \theta T1 & x1 & y1 \\ MT2 & \delta T2 & \theta T2 & x2 & y2 \\ . & . & . & . & . \\ MTn & \delta Tn & \theta Tn & xn & yn \end{bmatrix} \quad \text{.....(iii)}$$

To identify the best spot on the line, the variances of the spot position and the orientation are calculated with respect to the protein spot p in the source image. Let us assume that the length, angle and the mean variance are δ_v, θ_v and M_v respectively, then

$$|\delta_v| = \delta_T - \delta_s$$

$$|\theta_v| = \theta_T - \theta_s$$

$$|M_v| = M_T - M_S$$

The cumulative variance Δ is calculated as follows:

$$\Delta_i = (\delta_v + \theta_v + M_v)_i \quad \ldots\ldots\ldots\ldots \text{iv.}$$

where $i = 1$ to n and n is the number of spots.

The variances for each spot in the target image that lies on the line of path are calculated. The spot with least variances (Δ_{min}) is considered as the 'best possible matched' with that of the source image. The spot with the least variance (Δ_{min}) is considered to be the 'best matched spot'. The following approach is then taken to determine the variance of the shape contour.

ii. Shape contour variation:

Shape contour of the spot from the source image is approximated by the sample points M selected interactively (see Section 5.2.2) by the user. Every point of M can be described with respect to some reference point y inside the contour through a vector (Figure 5.4, Section 5.2.2). All vectors r and the points M are stored in a M-R table. An edge image I' is drawn using the values from the M-R table and the edge image, I', is mapped to the target region of interest of the target image. This is used to determine the shape of the target spot (Figure 5.11). The edge of the shape is used to determine the boundary of the target spot. The edge is determined by applying *Canny* edge detector (Canny, 1986). The pixel positions and r' are determined for the target protein spot with reference to the centre point in the region of interest (ROI). These values are stored in M'-R' table. The shape variation of the spots on the same line of path in the source and target image is compared by using the M-R and M'-R' table. The shape matching results are considered 'similar' if edge ρ of source image consisting X_i to X_{i+n} and edge λ of target image consisting X'_i to X'_{i+n} are within a given error tolerant limit.

5.2.4 Retrieving 3D structure of a protein

To map the protein spot with its 3D structural protein image, a dedicated, task-specific and declarative database (Raghavan and Garcia-Molina, 2001, Khan *et al.* 2002) is created to store the gel spot information. The set of spots in each gel electrophoresis is

labelled and each spot is associated with a finite number of values. A class is formed using these labels and values. Each entry in the class is of the form (L,V) where L is a label and $V=\{v_1,.....,v_n\}$ is a set of values. Each v_i represents a value that could potentially be assigned to an element E, if label (E) matches L. In our case element E is a spot which corresponds to the specific 3-D structure of a protein. An image feature extractor, M_f (Section 5.2), is used to look for the value v_i to initiate the search for the corresponding element E.

5.3. Experiments and Results

The following experiments have been conducted on a set of synthetic (created) and actual images. The results obtained from these experiments are correlated with the theories outlined in the previous sections.

5.3.1 Test dataset

A set of ten gel electrophoresis images are collected as test data from different laboratories (see Appendix D). These image data are available in public domain database 2DWG (See Chapter 4, Section 4.4). These images contain protein spots of plasma, liver, heart and serum sample of human. The images are all created using 2D PAGE technique but in different laboratories, using different protocols and under different disease conditions. However, all the protein spots in these images are represented in terms of molecular weight (X axis) and Isoelectric pH (Y axis) value. All these images are preprocessed before they are cited in the database. The size of these image files ranges from 250 KB to 2 MB depending on the nature and complexity of the images after the scanning (Table 5.1). The number of spots in each image varies significantly ranging from 300 to 2000 spots.

A set of five images (400x500) is created synthetically (see Appendix D) with the same background, intensity values and spot size to verify the proposed method (Table 5.1). A limited number of spots are created in these target images where a selected number of spots lie on the same line of path as it is in the source image. The synthetic image also ensures that the spot coordinates are determined selectively so that the neighbourhood search algorithm (Section 5.2.3.1) can be verified.

Table 5.1 Test Dataset

Type of Image	Name	Details
Real	AL4	Protein for Alzheimer's disease, used for target spot determination
	CSF	Human CSF, used as target
	PLT	Human platelete, used as target
	ELC	Erythroleukemia cell, image is used as source spot
	RBC	Human RBC, used as target spot determination
	HPG	Human hepg-superoxidase (HEPG2SP), used as target
	LVR	Liver fluid, used as target
	LYP	Human lymphoma cell, used as target
	SOD	Human liver SOD1, used as target
	LDL	Human ldl receptor, used for target
Synthetic	syn1	5 spots on line of path and 4 spots at random position
	syn2	4 spots on line of path and 4 spots at random position
	syn3	4 spots on line of path but not at same positional vector
	syn4	2 spots at neighbourhood and 1 spot at least variance position
	syn5	4 spots with different size and intensity value and 1 spot at nearest neighbourhood position

5.3.2. Identifying a spot on the line of path

Identifying the spots on the line of path in the target image requires deriving an array of mapping coordinates of the spots. Approach described in chapter 4 section 4.3.1.2 is used to determine the spots along the line of path. Nearest neighbourhood interpolation along the cross section of line is used to determine the intensity value of each point on the line. A threshold value of the spot intensity is used to select the spots. A threshold value of 150 is used for pixel intensity. The threshold values are evaluated according to the grey value distribution in the gel image. The grey value distribution of each image are determined empirically. The method captures all the spots which has the intensity value within the lower peak range of 150. Any intensity value exceeding this limit is considered as a noise or as a partial spot region. These spots from the line of path are not used as candidate spots for comparison. So only the spots with an array of

mapping coordinates are created to store the coordinates and the intensity value for each spot is found on the line of path.

5.3.3. Identifying a spot of interest in the target image

Experiment is conducted on the test dataset which is described in Table 5.1. The algorithm is first applied on the synthetic images (*syn1, syn2, syn3, syn4 and syn5*). These images have the same background, intensity values and spot size. A limited number of spots have also been created in these target images and a selected number of spots lie on the same line of path as it is in the source image (Figure. 5.7). The main objective of this experiment is to see if it retrieves the correct target image when it matches the selected spot from the source image to the corresponding spot in the target image. The spot positions are varied in the synthetic images. The orientation of the spots (angles θ_T) and the path distance (δ_T) of the spots are known. A real gel image, *ELC,* is used to compare with the synthetic image, *i.e. syn1*. Image *syn1* has nine spots and five of them are on the line of path. These five spots are candidate spots which are to be compared with the source image protein spot. The vectors of these spots are described in Table 5.2. When a spot is selected interactively in the source image it successfully retrieves the corresponding images when the orientation of the source and target spot matches. In this test the spot number five is the target spot because it has the least variance compare to other spots. This test is carried out on *syn1* and *syn2* to see if it successfully recognises the target spot from the number of spots. In each case of the experiment, it successfully identifies the target spot.

The next phase of the test is carried out to identify similar spots in the neighbourhood area assuming that an exact match has not been found in the first phase of the matching. In these tests, images (*syn3, syn4* and *syn5)* have been selected in such a way so that the exact spot corresponding to the source spot does not exist on the same positional vector on the line of path. In each case, it retrieves the target images which contain the neighbourhood spot that are considered as similar protein spots with regard to the source image protein spot. Four nearest spots at different orientations are used for each image (*syn3, syn4 and syn5*). When the test retrieves the target image, a region of interest is drawn on the target image. The orientation of the spots (the angle and distance

path) are recorded and they are compared with the pre-determined set of values. The result shows an accuracy of 1:8 match which indicates that it only picks up the best neighbourhood spot out of the eight possibilities (Figure 5.8). Same test is also carried out on other synthetic images to check if it can identify the nearest neighbourhood spot in the target image. In each case, the test successfully identifies the target protein spot at least variance position. The least variance is calculated as shown in Table 5.2.

5.3.4. Matching in 2D gel electrophoresis image

This phase of the experiment is carried out on real gel electrophoregram images. The gel electrophoregrams are not pre-processed to have unique background or to eliminate light spots which have higher grey scale values. A source image is chosen and a protein spot is selected which is to be matched with the same or similar protein spot in the target image. The line of path and the region of interest are drawn. The target image is then loaded and the matching operation is carried out to identify the same or similar spot in the target image (Figure 5.9).

Table 5.2. Vectors of each spot to determine the least variance

Spot	Angle	Length To spot	Mean grey value	Total value	Variance With source	Least Variance	Comment
Source spot	58.76	71.3	0	**130.0**			
1	28.36	128.4	0	156.7	26.6		
2	31.63	116.28	0	147.9	17.84		
3	39.49	95.9	0	135.4	5.34		
4	45.95	84.8	0	130.8	0.74		
5	58.76	71.3	0	**130.0**	**0**	**0**	**Spot Of interest**

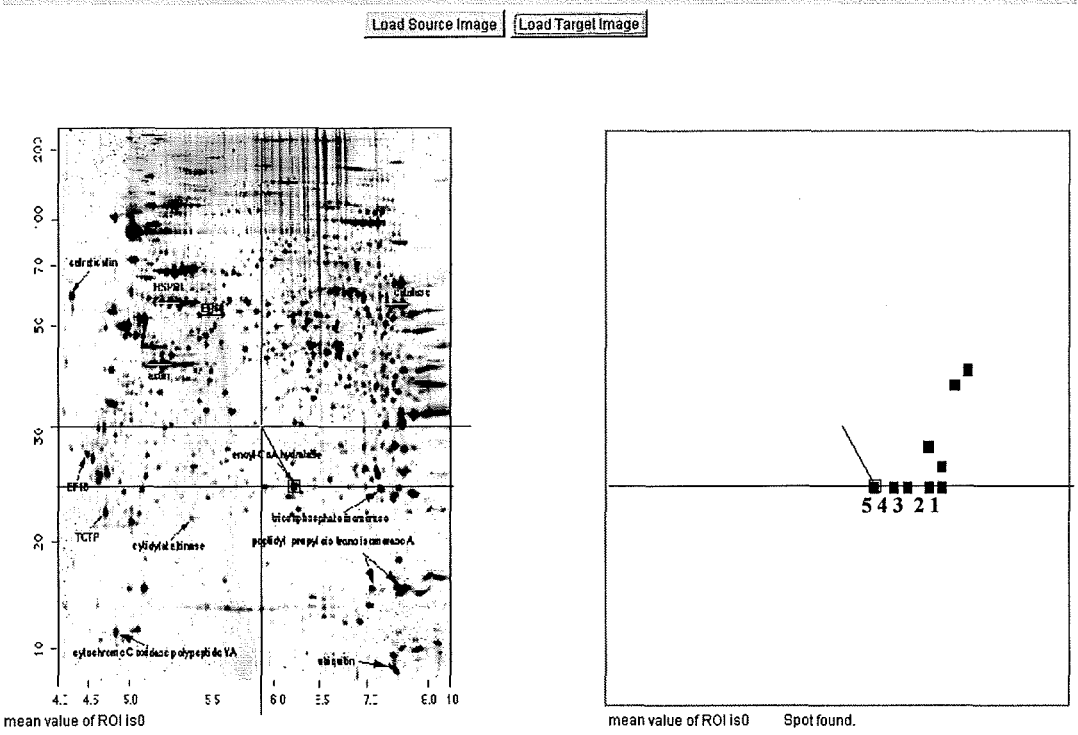


Figure 5.7. Identifying the spot at the same orientation as it is in the source image

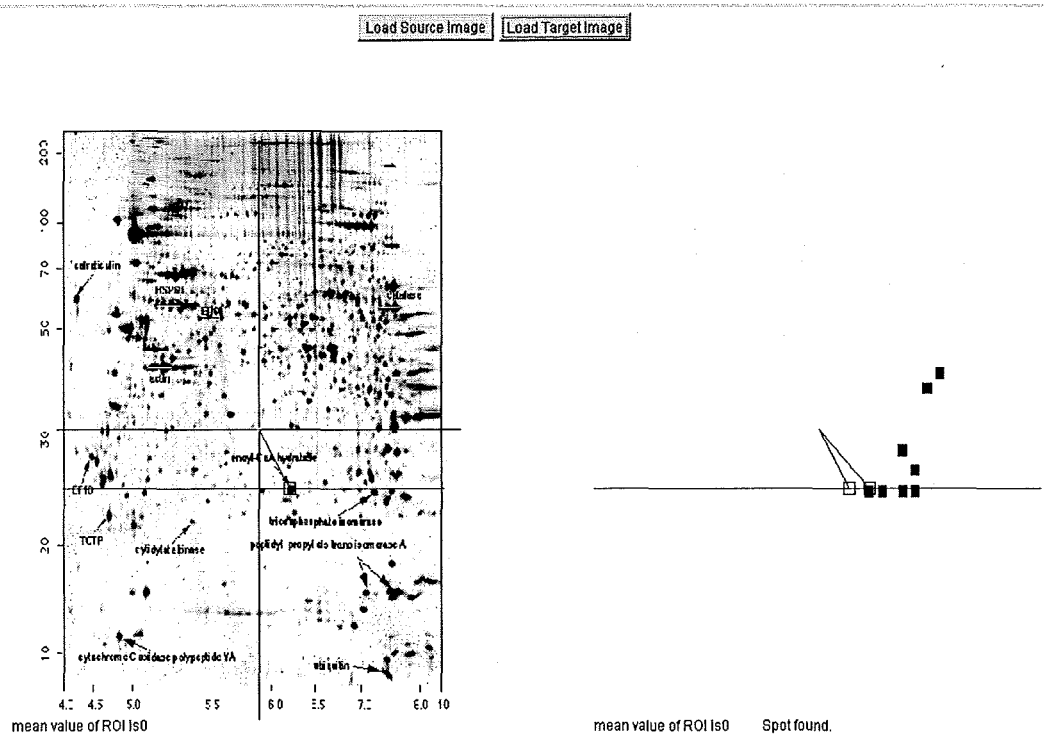
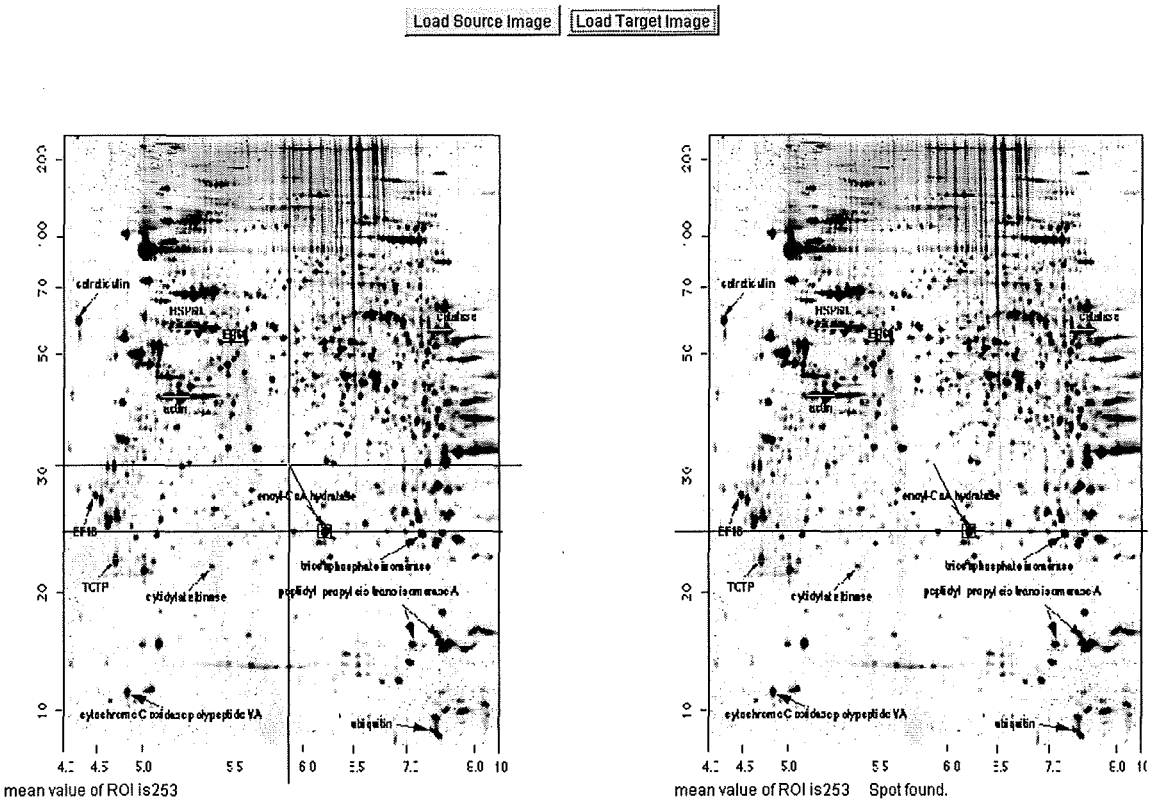


Figure 5.8. Identifying the neighbourhood spots at the least variance position



a b

Figure 5.9. Matching spot in the target image at the same location as it is in the source image. [a] Source image and [b] spot found in the target image.

The experiment is carried out on ten gel electrophoregram images (Table 5.1) and it successfully identified spot at the same orientation in different target images. In this case, the best nearest similarity value is used to choose the best matching. For similarity search, the variance value is used. This is carried out by determining the vector $\vec{v}$ and the variances Δ_i as described in equation (iv). The spot with least variance is chosen as the same or similar protein spot on the line of the path (Figure 5.10). Overall performance shows 90% (Table 5.3) successful identification of spots at least variance position in the target image. The image ELC (Appendix D) is taken as a source image. A spot is selected in the source image. All the target spots are identified successfully in the target image except the one in the LYP image (section 5.5). The image RBC (Appendix D) does not contain any target spot on the line of path and thus the algorithm could not find any similar spots as anticipated.

Table 5.3. Identifying the target spot in real image

Image	Angle	Length To spot	Mean gray Value	Total value	Variance with source	Spot type	Identified automatic-ally
ELC	58.76	71.3	0	130.0		**Source**	
RBC	-	-	-	-	-	No spot	Not on line of path
HPG	58.76	71.3	0	130.0	0.0	Target	Yes
AL4	56.44	73.1	0	129.5	0.5	Target	Yes
CSF	51.56	78.2	0	129.7	0.3	Target	Yes
PLT	59.23	70.1	0	129.4	0.6	Target	Yes
LVR	56.23	70.2	0	126.4	4.4	Target	Yes
LYP	56.87	70.2	0	127.0	3.0	Target	No
SOD	58.23	71.3	0	129.5	0.5	Target	Yes
LDL	61.34	69.0	0	130.3	0.3	Target	Yes

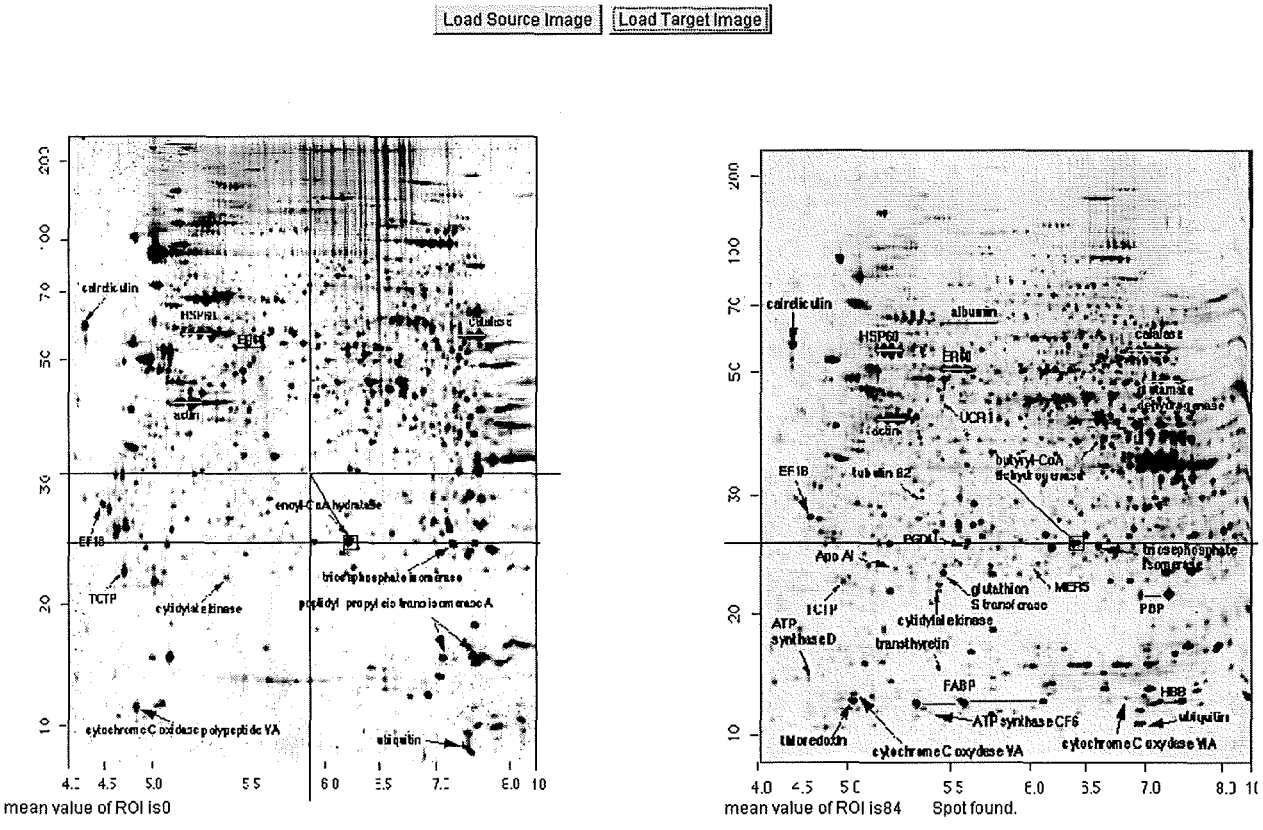


Figure 5.10. Identifying the neighbourhood spot the in target image.

5.3.5. Shape comparison

In this experiment a single spot is chosen and the variation between source and target spot is calculated to determine the similarity of the spots. The difference matrix is used to calculate the variance. For example, the spots in Figure 5.11 show the variation that exists between the two spots. Shape variance within an offset is taken as an indicator to establish that the spots are similar. Table 5.4 illustrates the parameters used for this comparison. These spots are considered to be similar because the shape variance falls within an acceptable region of tolerance value. In this case the shapes show 94% variance (Table 5.4).

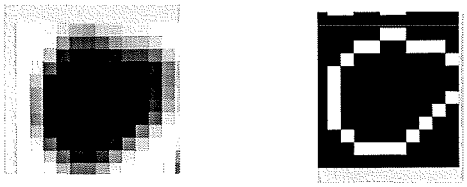


Figure 5.11: Source spot (a) and detected spot (b) in the target image for shape comparison.

Table 5.4. Determining the shape variance

Spot	x,y coordinates taken for comparison	*r* values for the coordinates	Average *r*	Shape variance %
Source spot	(5,2)	5.385165		
	(4,3)	5		
	(3,6)	4.123106	**4.68772**	
	(4,4)	4.242641		**94.04564**
Target spot	(5,3)	4.472136		
	(4,3)	5	**4.40860**	
	(3,4)	5		
	(4,6)	3.162278		

5.4 Retrieving 3D image

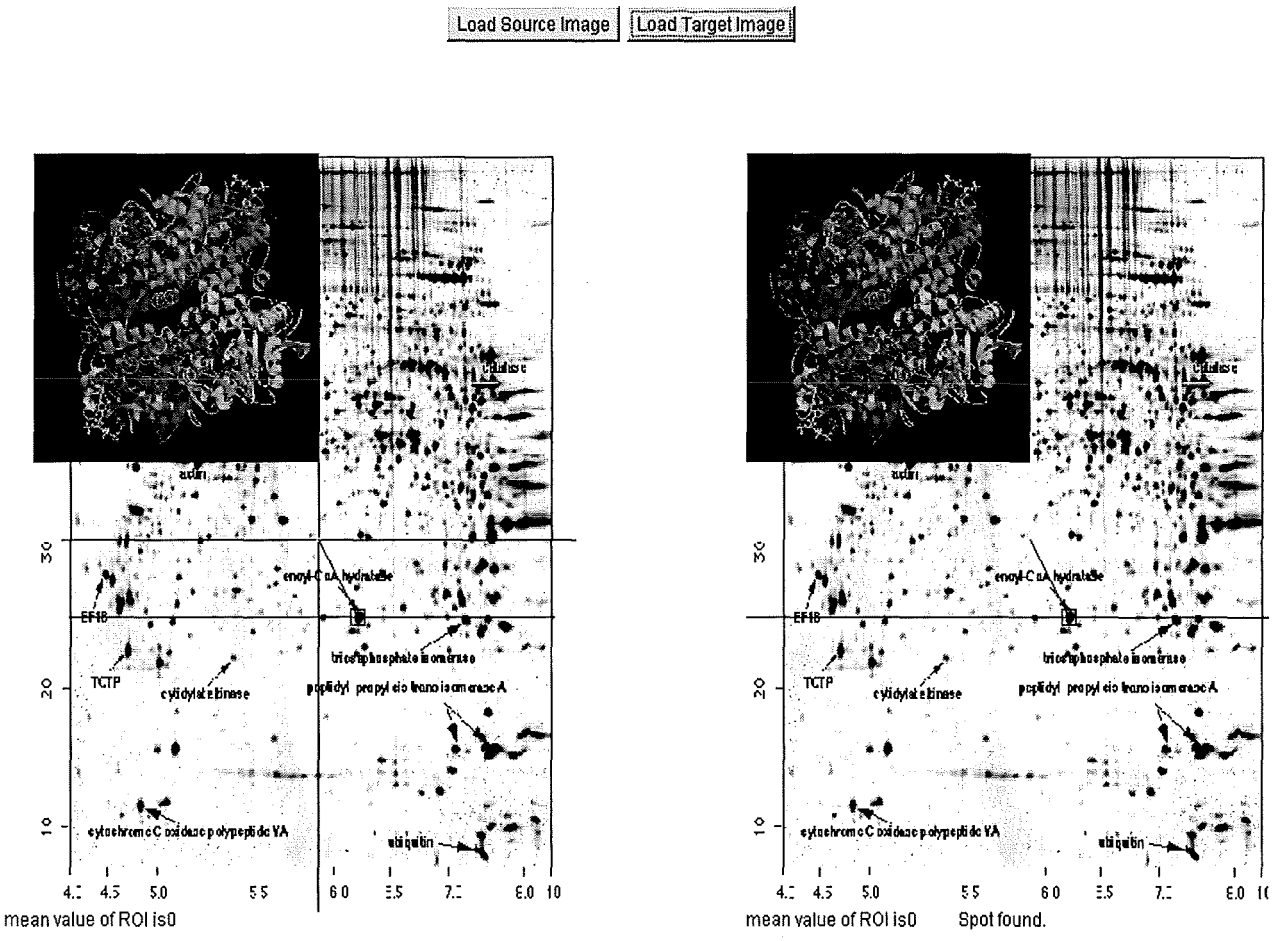
This part of the experiment is carried out to show that a 3D image can be retrieved interactively by selecting a specific spot from the source image. In this experiment, gel electrophoresis image spots are labelled using Melanine software (URL: http://us.expasy.org/melanie/). The following parameters are stored in the target specific

dedicated database: all the coordinates, intensity values, positional orientation and the average shape radius of the spots. When a specific spot is selected in the source image, the parameters of the selected spot are then matched within this database to retrieve the 3D image which corresponds to the selected spot. A dedicated and task specific database is created (Khan *et al.*, 2001) to store the resource mapping information. The gel resources are described in Resource Document Framework (Figure 5.12). The main feature of RDF is to provide interoperability by adding semantics to the web resources. RDF describes the whole web page or part of the web page as resource and these resources are named by URI. Resource URI with their properties and values are used to define a RDF statement. By defining the resources in the RDF body the researchers' will be able to configure their own choice of resource database mapping information according to their specific query. For example, to define a gel spot for protein *Oxidoreductase (Superoxide Acceptor)* in human liver sample gel electrophoresis image the following RDF statements are used to describe the metadata in dedicated database (Khan *et al.* 2003a).

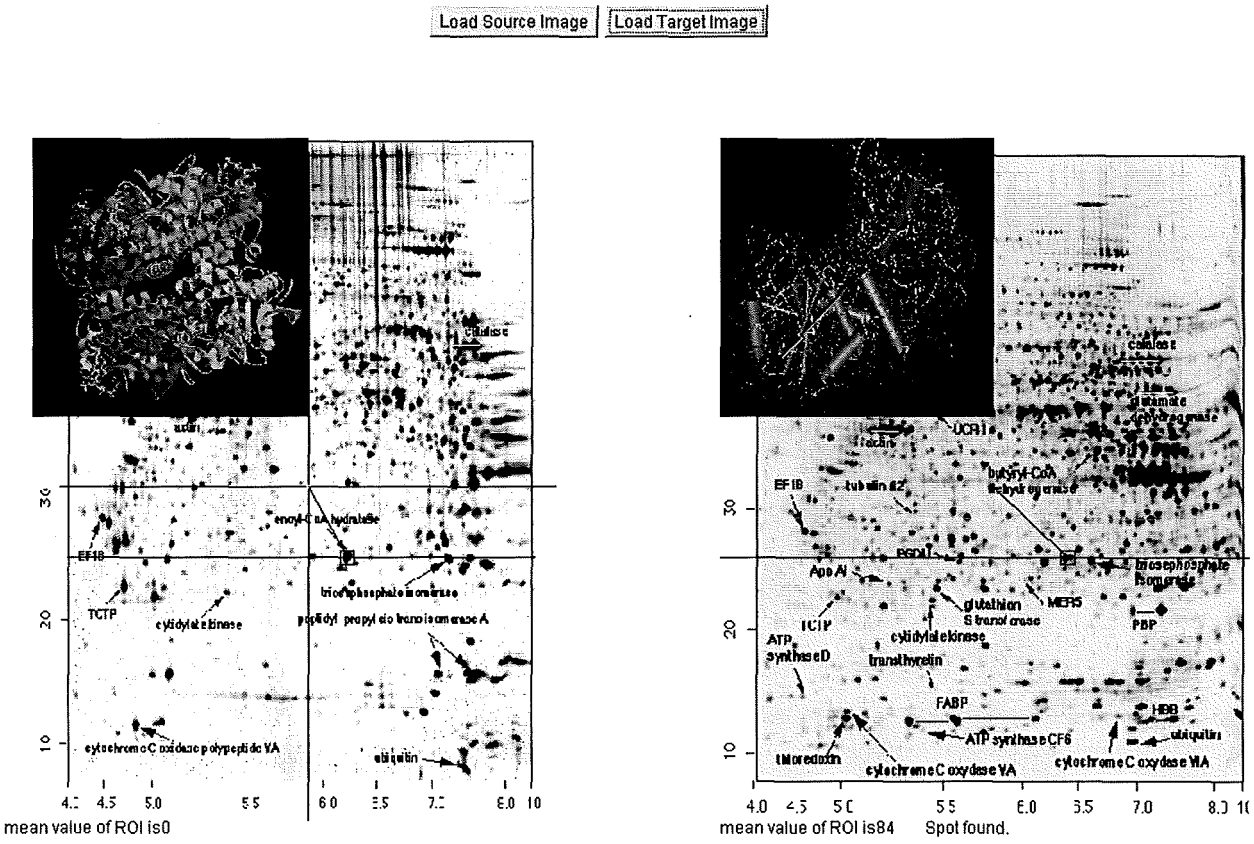
```
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:s=http://description.org/schmea/>
  <rdf:Description about=''http://www-lecb.ncifcrf.gov/cgi-bin/dbEngine/2dwgDB">
    <s:GelSpot1>
     <rdf:Description about="http://www-lecb.ncifcrf.gov/cgi-bin/dbEngine/2dwgDB
     <rdf:type resource=http://description.org/schema/Proteins/>
     <v:SpotLocation>
           <v:SpotDelta> </v:SpotDelta>
           <v:SpotTheta> </v:SpotTheta>
     </v:SpotLocation>
     <v:SpotFeature>
           <v:SpotIntesity>  </v:SpotIntensity>
     </v:SpotFeature>
     <v:ProteinDetails>
              <rdf:type resource= http://www.rcsb.org/pdb/cgi/explore.cgi?
              pid=92441066222151&page=80&pdbId=1SOS />
        <v:ProteinName>Oxidoreductase</v:ProteinName>
        <v:ProteinID>1SOS</v:ProteinID>
      </v:ProteinDetails>
   </s:GelSpot>
  </rdf:Description>
</rdf:RDF>
```

Figure 5.12 Description of gel electrophoresis protein spot using RDF

Figure 5.14 shows the 3D images that are retrieved from the local dedicated databases for same or similar protein spots.



a



b

Figure 5.14. 3D Protein structures retrieved from the dedicated database, [a] same protein and [b] similar protein for neighbourhood spot.

5.5. Summary

This chapter has presented a novel approach for identifying identical or similar spot from target gel electrophoresis image which lies on the same line of path as it is in the source image. A combination of geometric and image processing techniques have been used to identify the spot which matches with the features of the source image spot. Although the technique achieves significant accuracy in identifying the identical or the similar protein spot, some false matching were also resulted The following factors affected the final outcome in such cases:

i. image background

ii. the size of the spots, and

iii. the intensity of the spots

Performance can be increased significantly by adjusting the background contrast. Histogram equalisation can be useful tool for creating uniform image contrast. However, a nonlinear transformation can change the shape of the actual image. The contrast of the image background can also be adjusted by estimating the background values using morphological opening. Morphological opening has the effect of removing spot objects (circular shape) with a given predefined radius. But it also modifies the shape of the objects which therefore leads to image distortion. Binary thresholding and labelling can be used to pre-process the image for improved performance, however it can have a detrimental effect on the actual gel spots. Streaks, twin spots and complex regions can also have a significant effect on the identifying process. The performance can be increased in several fold by using the Brute Force method or LP approach (Efrat *et al.*, 2001) which can measure the ellipse encircling the spot. Also the technique described by Kriegel *et al*, (2000) to partition the streaks and the complex regions can have a significant effect when matching with the complex region. Ehrenmann *et al.*, (2000) used Generalised Hough Tranform method (GHT) for object shape determination.

GHT has been used for determining the contour of the shape and it has then been extended for shape comparison. A dynamic buffer of radius is used for creating the shape in the source image and the target spot boundary coverings are then checked with these values. The novelty of this approach is that the image has not been preprocessed in order to avoid the image distortion. The spot identifying process described here is dynamic and the image spot is selected interactively by the user (image object keying). The approach has emphasized on extracting the directional vector information from the image. Intensity value of the detected spot shape from the target image is also used as an additional parameter for achieving further accuracy. This helps to come to a conclusion about the spot similarity which enables to identify the protein similarity. The approach is also unique because it searches for the spot only on the line of path and it does not search the whole image. It uses the key concept of electrophoretic mobility of proteins. This concept reduces the number of candidate spots to be identified within the image.

The major objective of this approach is to retrieve the 3D structure of the protein from the Protein Data Bank so that the comparison does not only rely on the spot matching. Instead it is extended to the visual comparison of 3D structure of the protein.

To achieve this a meta data description for each spot of an image is created. The meta data is described in Resource Document Framework for its own flexibility. The elements of this meta data, delta, theta and the intensity value of the spot, are matched with the features of the image to retrieve further details about the protein. Once the source protein spot is selected the 3D structure of the protein is retrieved dynamically without any further intermediate keying. The retrieval of multiple variant protein structure can be achieved by describing the resources and assigning appropriate operator in meta data.

Next chapters, Chapter 6 and 7, will explore how this approach can be applied to initiate the retrieval of further details of the protein. The protein spot of the gel electrophoresis image can be further linked with molecular biology databases dynamically which are in public domain. The objective of linking the protein spot with other databases is to reveal further clinical and genotypic feature from the open resources that is associated with the selected protein spot in a gel electrophoresis image.

Chapter 6

Biological Database Integration: Concepts and Approaches

Chapter Objective

This chapter presents an approach to integrate biological data from multiple resources into a single page. The chapter looks at the existing integration approaches to construct federated information systems. The federated information systems development depends on creating mediators and wrappers. The wrapper creation for web data is complex and it is not accurate because of the lack of standardised and structured semantic of web data for molecular biology. This chapter explores the possibility of using context for integrating web data.

Chapter Contents

Chapter 6

Biological Database Integration: Concepts and Approaches

Computing infrastructures built to support the discovery process are often inadequate, and the potential of productivity gains through large-scale data integration has not materialised. While arguments rage about how this might have occurred, the reality is that, viewed as 'complete systems', life science IT lags nearly 20 years behind IT systems in other industries, where data are more structured.

Donnelly, (2003) Data integration technologies: An unfulfilled revolution in the drug discovery process, Biosilico vol. 1, No. 2.

6.1 Introduction

Integration of molecular biology depends on successful implementation of technologies and tools. In many research, for example, Cheung *et al*, (2001, 2000), Cluet *et al.* (2001) and Shanmugasundaram *et al.* (2001), these technologies and tools emerge as an aid to molecular biology database integration domain. However, database integration for molecular biology requires attention on a complete framework which is more than a technology. Gieger *et al.* (2003) highlighted in this regard:

The process of linking gene functions to the multitude of clinical phenotypes by means of information extraction is still in its infancy. There are a variety of concepts and terminology used in the clinical environment that are different from those used in genome research. Natural language processing methods have been developed to extract, structure and decode clinical information in patient reports. Nevertheless, the identification of relationships between entities in genome research and clinical phenotypes still remains a significant challenge.

Gieger *et al.* (2003) The future of text mining in genome-based clinical research, Biosilico, vol 1, No. 3.

Issues on framework and web data integration are discussed in this chapter. It especially focuses on how context based integration is feasible for web based integration. The issues that support this claim have been highlighted here. However, just to mention here that the complete methodology for designing, developing and

running a web based information system for molecular biology databases, is not the focus of this research and outside the scope of this thesis. The issues of any database integration which consists a number of layers and which holds mediators and wrappers for query execution and exporting schemas have been highlighted here. Section 6.2.1, 6.2.2 and 6.2.3 discuss the criteria for developing database federation techniques that are widely used for molecular biology integration. The issues of mediators and wrappers are also discussed here. Finally, a conclusion is drawn that further research for molecular biology database integration to concentrate on web data is required. Five critical issues for web based integration are then investigated and the drawbacks of the present approaches and methods are discussed.

Section 6.3 concentrates on the need of web based integration for molecular biology databases. This section discusses the necessity of metadata for web based molecular biology database integration. Although, the web developers created a strong connection between any search page and the database individually, the role of metadata was not established to represent a context or purpose of integration. The metadata plays an intermediate role to interconnect different objects residing in heterogeneous databases. These issues are discussed in section 6.3.1.

Section 6.3.2 illustrates the construct of metadata based on the context of integration. A metadata development based on subject domain entails describing a complete ontology. It has been argued here that a metadata based on context of integration considering an application domain of molecular biology eases the creation of the metadata and it helps defining the integration domain.

A carefully defined application domain in molecular biology leads to the development of wrapper and to resolve the semantic conflict among the objects. Wrapper also determines the navigational logic to execute queries and direct the queries to the target sources. The wrappers are the critical issue that needs to be implemented for web based data integration. Section 6.3.4 and 6.3.5 examine these issues and utilise these to implement the proposed framework.

Section 6.3.5 analyses the critical aspect of choosing language for describing the context in metadata. This is also another critical issue because many languages which are emerging to describe the data contents seem to cover only few aspects, *i.e.* either these are highly language specific or these are restricted to the use of describing their own ontology. This section is proposing that RDF should be used for describing contexts of integration and it highlights the reasons.

Section 6.4 finally summarises the chapter and establishes the ground for the proposed framework.

6.2 Integration Concepts for Utilising Multiple Biological Resources

The biomedical research laboratories are generating high volume of data each day. To manage these high throughput data, the organisations frequently create several different databases. These laboratory based data can be linked with federated databases for target information without creating its own global schema. Laboratory data will act as component of a database federation by adopting a component based approach for database integration. This will have a tremendous effect on the cost, efficiency and maintenance.

The federated information system can act as central point of access to a set of heterogeneous, autonomous, distributed systems. The laboratory database as a component of database federation will lead to the database resources which are to be integrated based on the integration domain. The objective of the biological resource integration is to interoperate the data between

- the structured, semi-structured and unstructured data sources,
- tightly versus loosely coupled integration, and
- database integration based on data semantics.

Local autonomy needs to be preserved for any integration of biological resources (Markowitz et al., 1996 and Karp, 1996). However, this is not a fundamental concern for business database integration where data is standardised according to business needs. The integration of biological resources requires to have the following features:

- able to access over the internet,
- able to navigate among different databases automatically, and
- not to be concerned with the structural knowledge of the schema.

This research is investigating the features of database federation to determine the suitability of current approaches for database integration. It later examines the approaches for database integration based on the context (semantics) of the web data.

6.2.1 Federated information criteria

Federated Information system is a collection of a number of autonomous databases which can cooperate with each other to exchange meaningful information (Davidson *et al.,* 1995). In Federated information systems the resources are mainly constituted by a number of databases with local applications around them and users

can have access to the global information system as a source of resources (Busse *et al.* 2000 and Sarker *et al.,* 2003). Unlike the Federated Database Systems, basic characteristics of Federated information system is based on the fact that participating resources are not restricted to database systems. This can include a wide variety of information provisions.

The architecture of federated information system is based on the integration of global schema. This global schema can be integrated by exporting schema from participating data resources which is termed as 'bottom up strategy' or by defining global information requirement and developing a global classical schema which is termed as 'top down strategy' (Busse *et al.,* 2000). In bottom up strategy the integration process starts with the analysis of horizontal correspondence between component schemas. To implement this, an integrated schema is derived together with the correspondences between the integrated schema and the export schemas.

In contrast, the top down strategy consider the existing information resources and then generates a global schema using formal analysis process. In this strategy a vertical correspondence is established between the global schema and source schema to allow the translation of queries. Up till now the Federated information systems research is within the scope of 'tightly coupled' approach where a schema export environment is created and consolidated to one global schema using either top down or bottom up strategy. The architecture of tightly coupled Federated systems is shown in Figure 6.1 which was proposed by Busse *et al.,* (2000).

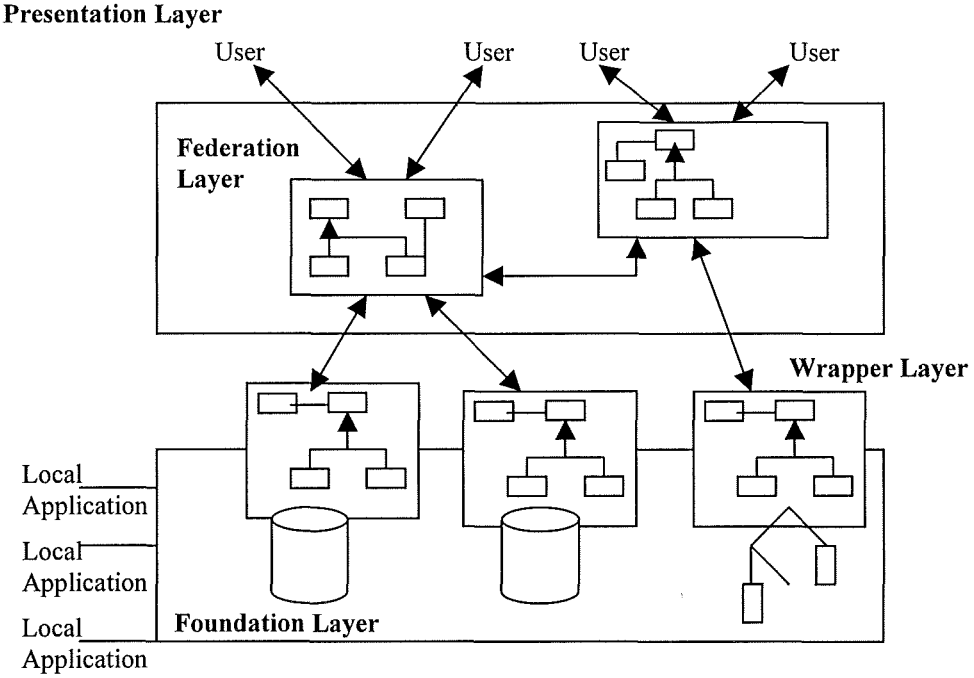


Figure 6.1. Architecture of a Tightly Coupled Federated Information System

However, tight connection between federation and foundation layer greatly affects the incremental development of the system and any changes in the schema of foundation layer or in the export schema trigger a new integration process.

Busses *et al.* (1999), classified the Federated information systems into three different groups. A brief description of these groups are given here.

- Any federated system without the global schema is called 'loosely coupled', and the federated system with a global schema is called 'tightly coupled'. Tightly coupled approach oversees information resources as a single organisational entity, whereas, loosely coupled approach shares a common syntax for data publishing and querying (Kemp *et al.*, 2000a). 'Tightly coupled' approach adopts common hardware and software for participating sites. In loosely coupled approach user do not need to know the individual query language, however, the user does need to know the schema of the participating databases for writing the query.
- Federated systems which are permanently materialised of source data in one local space showed a widespread acceptability in molecular biology database domain (Schonbach *et al.*, 2000; Paton *et al.*, 2000). This approach is termed as 'Data Warehousing'. Although, the warehouse approach is very fast in data searching, it has high risk in schema conversion for data materialisation in molecular biology. Furthermore, it has a risk of suffering with outdated data because of continuous update of participating molecular biology databases (Markowitz *et al.*, 1996).
- Any federated system with lesser extent of semantic integration can create a major challenge for extracting meaningful information. A higher degree of integration is achieved if only results from different sources are merged and if they correspond to a specific context for molecular biology. Identification of objects is necessary to achieve this, however, this is difficult for molecular biology databases because of the semantic conflict and lack of presence of context for integration (Karp, 1996b).

6.2.2 Mediator for database federation

A federative database approach for molecular biology database integration is not explored in detail. Kemp *et al.*, (2000a) attempted to create database federation of molecular biology resources using mediator for database federation. In their approach

mediator describes the content of the data resources that are members of the federation, including the semantic relationship which exists between them. This mediator also maps between mediator's schema and the external resources' schema.

A mediator based integration is a 'tightly coupled' integration for database federation. It follows a top-down approach for creating database federation. The creation of mediator follows the following steps:

i. Integration requirement needs to be analysed according to the different levels of users,

ii. It leads to different views because different users have different requirements, and

iii. The resulting view is homogenised immediately by integrating the participating views (Navathe *et al.* 1986).

A mediator based integration in database federation offers some advantages in maintaining the system. A mediator also allows changing the structure of data resources without affecting the global schema, as it only requires to change the correspondence mapping (Laser, 1998). But, it faces the problem of finding a suitable global schema.

The top-down approach also leads to a less coherent system since they introduce only weak bindings between the sources and the integration is based on common ontologies. In this strategy a comprehensive description of concepts and specifications are created on a particular domain and this ontology is then used for describing the content of the data sources.

A database federation consists of more than one mediators. All the mediators have their own schema. One schema can relate to other schemas of other mediators. This leads to a high risk of structural and semantic conflict.

Other research in homogeneous mediator based information system, Von (2000), used the same data model and query language for all the mediators and data is structured. This creates a major challenge in dealing with the heterogeneous data sources with heterogeneous mediators which is essential for molecular biology database integration.

6.2.3 Wrapper for database federation

A wrapper transforms data that is represented in one particular data model of its "wrapped" data source into a data model representation of the mediator. It also

translates queries from the query language of the mediator to "queries" executable by the source. The schema for wrappers depends on the sources as it is a source specific operation and on the interface that the wrapper uses to access the source. But it is independent of the mediators schema. A wrapper can be built at the site where data is resided or at the site where mediator is resided (Von, 2000). A wrapper can be reused for different data sources of same type, for example, a wrapper for RDBMS of one data source can be used for another data source with the same RDBMS.

However, problems arise when integrating a new source into the global systems for the following reasons: (i) specifying a new service based on the knowledge of the information need on top for creating view, (ii) specifying the knowledge of the available contents at bottom for local databases and (iii) relating these to each other. These also require to modify the mediators or to create a new one. Moreover, these might lead to create new architectural components in the integration layer (Kutsche and Sunbul, 1999).

The problems rise to several fold when many Bioinformatics or Biomedical organisations develop and maintain independent databases. These information become meaningful if they become part of complete dataset, such as web based resources for Protein Databank or Genome Databank. Web based resources are the primary issue for the future of molecular biology database integration. Generating a wrapper for these web resources is a major challenge. Wrapper for web based data resources are difficult to reuse and it also lacks the standardisation. So the challenge is to generate a wrapper easily and efficiently.

The next section describes the web data as information resources for molecular biology and the approaches taken to integrate these web resources.

6.3 Web Based Data Sources

Molecular biology data are emerging as web based resources. These web data are open, semi or unstructured and hyper linked with one another with different extents of semantics. These sources often contain overlapping or complementary information, but they have their own semantics heterogeneity of relationship which is to be hyper linked. So, the challenge of integrating these biological web resources lies in extracting and synthesising information from multiple independent web sources (Huck *et al.,* 1998).

The problem of semantic heterogeneity is to identify the semantically related objects in different databases and to identify the resolution of schematic differences

among them (Kashyap and Sheth, 1998). The web deals with unstructured data which is not standardised by any authority and this web data is also difficult to exploit as it lacks the semantics of its contents. To develop an integration domain and mapping among the objects residing in multiple sources is a challenge. In addition providing semantics on the contents for data interoperability can make the web based data into a good source of knowledge (Sondag, 2001).

The next section explores how the Metadata, Context and Navigational approach among the resources are used for conceptualisation and for adding semantics to web contents for its interoperability.

6.3.1 Metadata for webs

Metadata is defined as data or information about data. Metadata is one of the pivotal ideas on which the database components depend. The function of the metadata is the ability to abstract and capture the essential information from the underlying data which is independent of representation details (Kashyap and Sheth, 1998). A general classification of metadata according to Kashyap and Sheth (1998), is described in the following Section.

6.3.1.1 Metadata types

Various types of metadata are used by different researchers. Two main types are Content Independent metadata and Content Dependent metadata.

Content Independent Metadata - this type of metadata captures information that does not depend on the content of the document with which it is associated. Information content is not captured by this type of metadata but this might still be useful for retrieval of documents from their actual physical locations.

Content Dependent Metadata - this type of metadata depends on the content of the document it is associated with. Examples of this are size of a document or date of last update. The content dependent metadata can be divided into following two categories.

Direct Content Based Metadata - this type of metadata is directly based on the contents of a document, for example, full text indices based on the text of the documents.

Content Descriptive Metadata - this type of metadata describes the contents of a document without directly utilising the contents of the documents, for example, textual annotations describing the contents of any page. This specific type has two sub-categories:

Domain Independent Metadata – in this case metadata captures the information present in the document which is independent of the application or the subject domain of the information, for example, parse trees for HTML or C++.

Domain Specific Metadata - this type is described in accordance with the application or subject domain of information.

Molecular biology researchers deal with consolidated view of data for any particular subject or application domain, for example, integration based on human disease data, integration based on mutation data or integration of data for cell developmental issues. Domain specific metadata is a major issue for web based molecular biology integration. Contents in a particular web for molecular biology only represent a partial datasets within a particular domain. This dataset is not complete unless it can be linked with other web pages for different sets of data to give it a more meaningful information within that particular domain of molecular biology. For example, OMIM web page describes diseases but it does not provide complete information about gene and its products. But, when it is linked with GDB or PDB web pages then it gives a complete information about the gene and its products. It will also give information about any disease that might be caused due to the mutation in the gene. However, this integration is only useful for the study of disease but it is not useful for any developmental study. One needs to create another domain specific metadata describing the contents of the web for such study.

This implies that domain specific metadata creation is content dependent of the web that it represents. It helps to abstract the representation details for a particular subject domain which are more meaningful. Domain specific metadata does not rely on underlying structure or organisation of the data. It takes into account the context of integration which might vary from domain to domain. Therefore, this is the most suitable metadata type to deal with semantic heterogeneity of web which is open and unstructured.

6.3.2 Contexts for integration

A context is a knowledge that is required to answer a query for another system. This can be achieved by the mapping between different schema elements (Ouksel and Naiman, 1993 and Rector, 2004). This context represents the meaning, contents and the properties of data. A metadata is associated with this data for representation (Rector, 2004).

To understand and represent the knowledge of any web or a collection of web pages it is required to capture the context of linking the objects and it is also required to map the related domains of the two objects.

Kashyap and Sheth, (1996 and 1998) identified context which involves a group of databases and their relationships between the objects of different entities in a particular subject domain.

A context can describe and abstract the representational details of the underlying data. These contexts are usually consulted first before processing the underlying data but they can also be consulted at run-time. These type of contexts represented in metadata are called Metadata Context (Kashyap and Sheth, 1996).

Other type of context which captures the domain knowledge rather than the underlying data itself and which forces to represent the conceptual semantic view of the underlying data is called Conceptual Context (Kashyap and Sheth, 1996). This conceptual context is built on the terms that are used within a specific domain. The terms are required to interrelate with each other which depends on the relationship and which have been described in the subject ontology.

The conceptual context can provide a very useful mean to tackle web semantics in molecular biology database integration. The challenge of building this type of conceptual context in molecular biology is to bridge the gap between terms used in one subject domain and the terms used in another subject domain. Therefore, a relationship mechanism needs to be established within a particular subject domain in molecular biology.

The relationship among the terms used in web based molecular biology databases varies from one subject domain to another. For example, Nucleic acid used in Protein Databank is related with Gene in Genome Databank, but the meaning is the same in both cases. The interest of users which depend on the organisation or groups can also overlap. For example, the users who are interested in diseases will be looking at the protein data and its structural perspective, whereas the users who are interested

in cell development will be looking at the protein data and its development perspective at each stage of the cell.

In molecular biology database domain a fixed set of descriptions of relationships between objects does not necessarily mean the semantic similarity between them. An example in this context is explained in the previous paragraph.

Currently web data are link-based, *i.e.*, connected via links, and abstracting links from documents allows great deal of flexibility. Link abstraction provides the addition of functionality feature to the web data and it does not affect the conceptual context. It also simplifies the link maintenance by reusing the links in different domain. This is why the research has used the conceptual context idea for molecular biology web data integration

6.3.2.1 Using context in web data integration

Kashyup and Sheth (1996) formalised the definition of objects and their association among themselves within a context scenario. Any object in a context scenario specifies the assumptions and helps to extend the object within any database federation.

A context can be used as a set of contextual coordinates (Kashyap and Sheth, 1998) and it can be represented as, context = $<(C_1, V_1)\ (C_2, V_2)....(Ck, Vk)>$

where each *Ci* corresponds to a context and each *Vi* represents the relationship between the objects for the context. Context represents the role of the objects and why they are related to each other in a particular context of subject domain. The concept of context can be applied to web data integration for molecular biology where the role of *Ci* and *Vi* can be explained as follows:

Role of *Ci* :

- *Ci*, $1 \leq i \leq k$ is denoting an aspect of context which is a member of the subject domain.
- *Ci* models the subject domain in a context to link with other aspects of the context.
- *Ci* may be based on the assumption of links which are allowed by the subject domain. Any relationship which is not supported by the subject domain is null and void.

- *Ci* may not be associated with an object *O* in the web data although it is in the subject domain. It does not associate with the application domain or outside the boundary of the users interest domain.
- *Ci* will be associated with *Cj* $(1 \leq i \leq k)$, $Ci=<Cj,Vj>$ when and only when $Ci \in C$ and $Cj \in C$, where *C* is a unifying context of all contexts.

Role of *Vi*:

- *Vi* can be expressed as a set of values for a particular context.
- *Vi* can refer to another object of another context for relationship mapping.
- *Vi* can act as a value buffer to extract appropriate values from the web based databases that support an aspect of context.
- *Vi* can unify the values that are collected from different databases in their respective contexts when it is referring to a unifying context.
- Unified *Vi* can be referred as value extensions for expressing any object more explicitly that support an application domain.

El-Beltagy *et al.* (2001) and Mork *et al.* (2001) proposed a framework for adding links to web pages based on the context of the user and the web pages. They used a simple rule based algorithm to utilise links in the creation of generic links in context. They proposed that if document X and document *Y* appear in context *Z*, and if there is a link related to a concept *C* in document *X*, then the same link can be applied to concept *C* in document *Y*. This applies to all documents in context *Z*. Thus the model allows to extract multiple source nodes and multiple destinations. Their algorithm was based on Term Frequency and Inverse Document Frequency technique where they measured cosine similarity function. In their proposed technique they abstracted a context *via* a cluster centroid of terms which represents any given cluster to group the documents. The major drawback of this technique is that if multiple concepts represent the same centroid of terms for similarity matching, then it is unable to differentiate between the documents which represent different concepts. This scenario is quite common in molecular biology data. For example, gene sequence annotation and genotypic details of any particular trait share the common terms for concept representation.

This idea can be extended further by conceptualising the contexts to design any object and to assert how the components of the objects are distributed in multiple web sources. These assertions can be described with the modelling of appropriate

contextual coordinate, which is a particular web resource of interest for particular subject domain in this research. Although, it may hold different aspects of subject domain, by associating with other objects of other aspects it can give a new object for a particular application domain (Figure 6.2). The parent context comprises a set of component contexts and values. The component contexts describe a set of objects which are a subset of any application domain. The objects which are described by the component context are interrelated with each other to define a unifying context for integration purpose.

A group of similar contexts with their associated values are grouped together in a set which becomes a Parent context for all the contexts in the group. Parent context which is a unifying context can hold different objects which is true for their respective contexts and application domains but it might not hold a complete set of objects for any subject domain. Mapping of the objects in different schema for web can be described by Interschema Correspondence Assertions and a set of Intershcema Correspondence. Assertions reflect the Parent context. In our approach it is integration domain for a particular application domain (see Chapter 7 for more details).

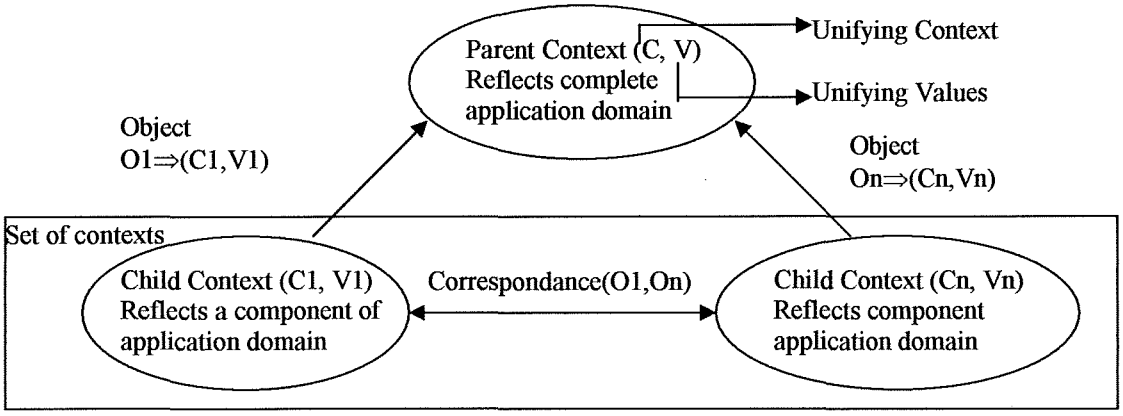


Figure 6.2 A simple schematic diagram of context components

This research has described context as a collection of relationships where any object which is preloaded will initiate the context activation. These collection of relationships will also provide a number of access methods to the target web data and navigation based on the context map (Chapter 7; Section 7.3.1). The research has also suggested a novel navigational method and an integration domain for accessing web based molecular biology data.

6.3.3 Wrapper for web data

Web is a vast source of molecular biology data. In most cases these data are hidden behind the search forms. These searchable data resides in structured and unstructured databases. The pages for any web queries are dynamically generated when a query is submitted to web search forms. It is a major challenge for any web wrapper to automatically parse, process and interact with the form-based search interfaces. The steps involved in submiting a query and getting a response are outlined in Figure 6.3 (Raghavan and Garcia-Molina, 2001).

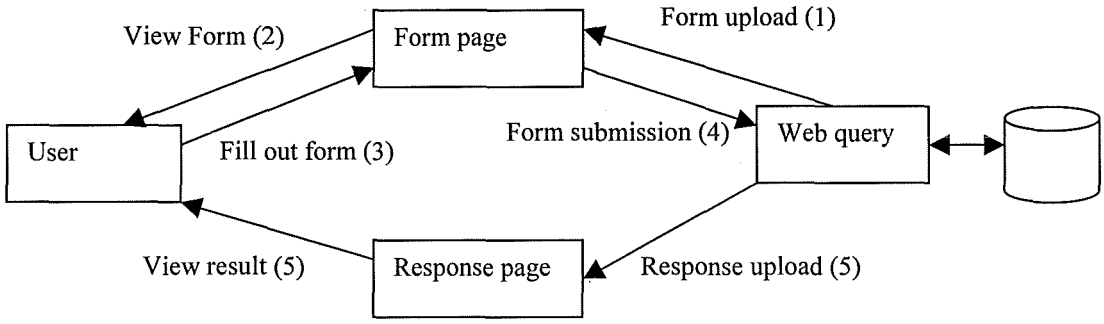


Figure 6.3. Steps involve in web queries.

The major concerns for web based data sources to extract any relevant information are:

- wrappers need to have fault tolerant parsers to collect any desired information
- wrappers need to divert the searching operators to a number of desirable sites, *i.e.*, it should be able to link with other navigational site logic so that it could reach the destination and collect the relevant information from the HTML pages.

It is time consuming and tedious to build wrapper from scratch if no tools are involved. So, it is quite reasonable to build any wrapper using Wrapper Specification Language (WSL). WSL uses text parsing using regular expressions of grammar rules for HTTP operations. These operations include, form submission, requesting a web page and parsing the HTML pages. Some of the examples of this language are JEDI (Huck *et al.*, 1998) and W4F (World Wide Web Wrapper Factory) (Sahuguet and Azavant 1999 and 2001).

In Sahuguet and Azavant (2001), the authors describe the wrapper language W4F as a toolkit to generate wrappers for web sources. W4F functions as an environment which allows to generate wrappers using a declarative specification

language and compiles it as a Java component. This Java component can then be used as a part of an application.

An interface between mediator and wrapper acts as denormalised schema or universal schema to export information to the web source and to extract any information from the web. Mediator represents a view of the data to the client once a successful extraction is completed. W4F performs the following four major tasks:

a. Retrieving a web document
b. Cleaning
c. Extracting information, and
d. Mapping information

Web wrapper consists of three layers. These layers are:

i. Communication layer
ii. Extraction layer, and
iii. Restructuring layer

The retrieving, cleaning and extracting, and mapping functions correspond to the communication layer, extraction layer and restructuring layer respectively. A general architecture of W4F wrapper as described by Sahuguet and Azavant (2001) is shown in Figure 6.4.

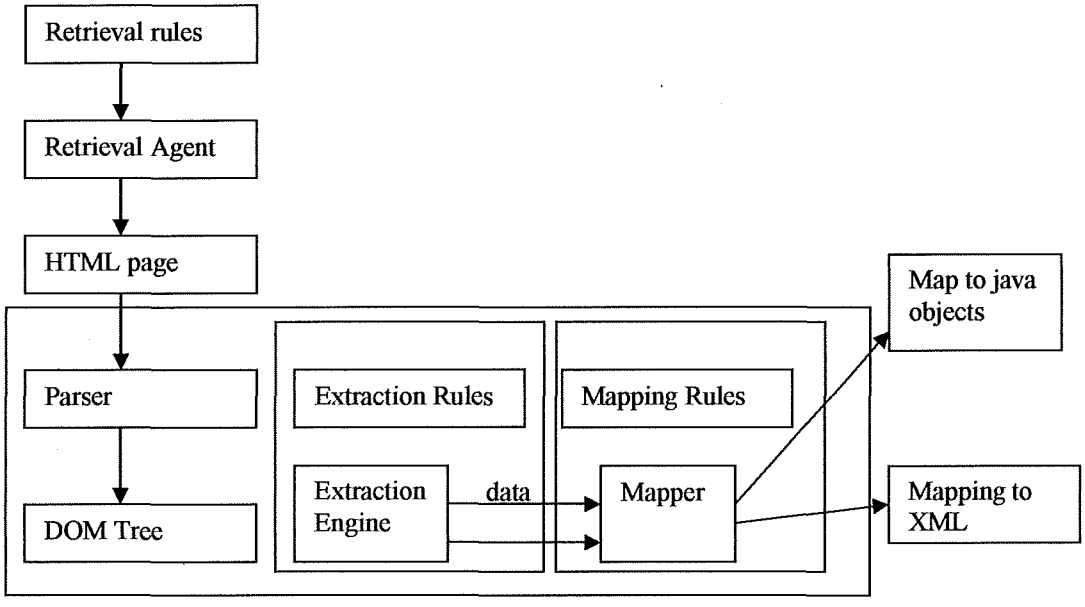


Figure 6.4. Block diagram of W4F architecture.

Once a page is retrieved using HTTP protocol, the cleaning stage transforms the HTML document into a well-formed document so that it can be parsed into a Document Object Model (DOM) tree. A set of extraction rules using HEL (HTML Extraction Language) plays a vital role to traverse through the DOM tree to collect the

elements. In the next stage the mapper maps these values to an exportable structure which will be suitable for any specific application. Manolescu *et al.*, (2001) used a similar approach. They used DOM tree to convert multiple structured documents into XML repositories and it allowed the users to make queries on these repositories.

However, searching web data in molecular biology database requires to transmit the queries to the wrapper in the form of binding attributes. These queries then need to be pushed to the source by invoking the search form created by the source web sites. For example, Protein Data Bank (PDB) web site has details of protein and nucleic acid structures and other phenotypic details. One needs to fill a search form to make a query in this web and the form is then submitted to find the required information. The database can be searched by using keywords, abbreviation or by using accession number. Wrapper passes any one operator at any one time to perform the search and converts the required HTML page into DOM tree (DOM, 2000) for data extraction. It also relates the operator with other operators to find any specific source and to push the query in the source at the same time. For example, accession number for GDB needs to be related with the accession number of PDB in any given context, *i.e.* linking genotypic information with the phenotype for disease analysis. In general, web based data is accessed through the wrapper by using HTTP. Von (2000) suggested two essential mechanism for web based access:

i. Request HTML pages and execute 'CGI programs'

ii. Web forms trigger the query execution

He also proposed a very general architecture (Figure 6.5) to represent any wrapper for web based data access.

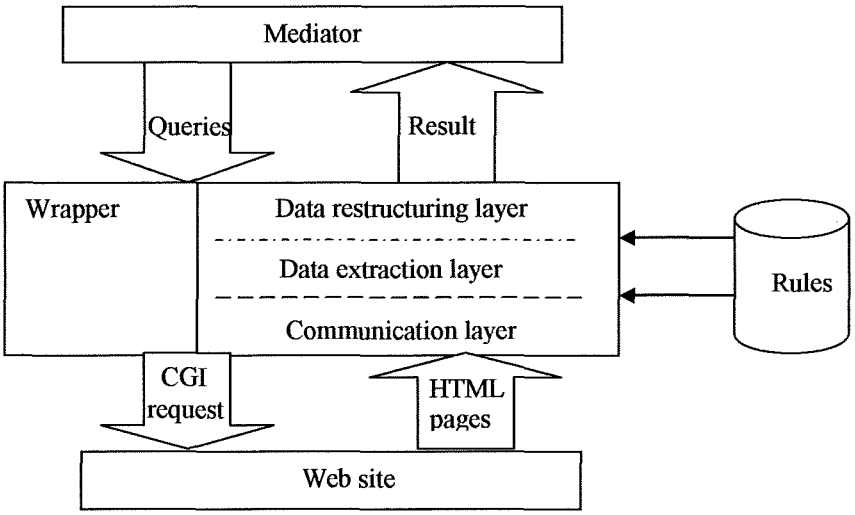


Figure 6.5. A general architecture of a web wrapper.

Shaker *et al.* (2002) also adopted the concept of utilising the semantic mapping rules for transforming source data space into mediated schema space. They stored the rules externally to the metawrapper application.

This research used Von's (2000) approach to access the web search form and to retrieve the required HTML pages. This is described in Chapter 7.

6.3.4. Navigation through the web sources

Present web based molecular biology database sites are all form based (Freidman *et al.,* 1999). They do not follow a well defined topology of HTML pages. They manually operate on large databases which are updated on a regular basis. These databases can be searched by using the search forms and by giving some search criteria. A HTML page is created dynamically in response to the query form submission. The web pages also allow navigation through multiple servers where pages are cross linked. For example, PDB and SWISS-Prot (Bairoch, 1993) servers are cross linked for navigation. The pages are also cross-linked with other pages by hyperlinks or queries. These queries can be executed by using a hyperlink which executes a script located at the resource server which retrieves the required static page.

The success of any web based database integration depends on whether the data of the resources can be located and accessed dynamically. However, existing techniques for web based integration mainly focus on the sites which are composed of conventional static pages. The commercial tools and mining prototypes for the web are also designed for these types of conventional sites. The enormous amount of information stored in database and archives have not yet been explored using such dynamic query (Berendt and Spiliopoulou, 2000).

Berendt and Spiliopoulou (2000) highlighted two issues regarding the quality of navigation for form based web sites. These issues are:

- support for navigation across the generated pages which are resulted from the retrieving documents and which contain links to other documents, and
- query capabilities with different searching and browsing patterns, *i.e.* key work searching, submitting accession number of any particular object which may exist within the interactions of a given web site as a result of users needs and interests.

Although the individual sources support a common semantics, for example, shared meaning for the data types and domain, sharing the data among the global community of users is still a difficult task. Data sharing for the global community is facing the following challenges (Mihaila *et al.*, 2002):

i. data providers need a mechanism for describing and publishing available sources of data

ii. data administrators need a mechanism for discovering the location of the published sources and they need to be able to obtain the metadata from these sources

iii. users need an effective mechanism for browsing and selecting sources.

In MEMOIR (Managing Enterprise-scale Multimedia using an Open Framework for Information Re-use) project Roure *et al.* (2001) attempted to support the collaboration effort in an Intranet environments by sharing documents. They employed one or more link servers which can be interrogated either before or after a document is viewed. However, MEMOIR is highly dependable on organisational infrastructure and applications.

A comprehensive navigational approach is required to locate and access the molecular biology data sources. The efficiency and the accuracy of such navigational plan also depend on the relevance of data that have been extracted. Such navigational plan on the basis of application domain is proposed in Chapter 7, Section 7.4.

6.3.5 Defining contexts and relationships

Apart from the HTML documents both XML and XML Schema have also been adopted widely for molecular biology repository. Wong and Shui, (2001) and Cheung *et al.*, (2001) described the integration using XML database approach and XML applications respectively. WWW consortium also proposed the Resource Description Language and a mechanism for resource location. The example of this is the emergence of Resource Document Framework (RDF) (RDF, 1999). However, there is still little support for describing data resources for web and finding the sources using metadata for the resources. But at present, the researches are still focused only on bibliographical collection and on information sharing among specialised collection (Mihaila *et al.*, 2002). Therefore, attempts are made to add semantics on data objects for defining the contexts for the integration purpose.

Despite the impact of HTML and World Wide Web on distributing data, the main limitation of web is its lack of machine-understandable semantics. This limitation does not allow to extract the concept of relationships from these distributed data or to correlate the data with each other (Halevy *et al.*, 2003).

Attempts to focus on web semantics are carried out by turning the web into knowledge. First step to convert the web into knowledge is to define the meaning of the data and its relationship. Knowledge representation language RDF and RDF Schema allow to define different domains and to relate them by using reasoning about the concepts (Appendix C). It is evident that no domain posses a single description to represent domain ontology, instead it requires few overlapping descriptions to represent the concept of any particular domain of subject. This research proposes to determine the application domain of integration so that single description can be used to fulfil the requirements and the description of concept can be represented by RDF or XML for interoperability.

In Halevy *et al.* (2003), the authors reported about two significant problems to represent the language for developing any concept. These problems are:

(i) A vast gap between RDF and data providers: RDF represents everything as a set of classes and properties and it creates a graph for relationships. So, RDF is focused on identifying the domain structure. The data providers are generating information which does not follow the complete ontology and the data are based on the application domain. So it is required to fit the data into the RDF for complete representation of the domain structure

(ii) Exporting data into XML: many data providers export their data into XML (Shanmugasundaram *et al.* 2001). XML has limitations to define the domain structure, instead, it tends to define the most important object or entities. So, it provides very comprehensive hierarchical details of the objects and leave the relationship among the objects unspecified.

This research is proposing to use RDF to represent the context of the relationship because:

- RDF does not emphasises on object's importance, it plays a neutral role.
- It represents graph oriented relationship to link the objects, properties and values.

- It uses references, containers and certain properties and thus adds semantic meaning to the objects.
- Relationship between the objects can be defined explicitly with names.
- In contrast to XML the relationship can be conceptualised in RDF description.
- RDF can name all binary relationships between pairs of objects, whereas, in XML objects are embedded in hierarchy within the document structure.
- RDF allows the representation of semantic meaning in RDF Schema or it allows the representation by simply interpreting the data.
- XML schema is based on object-oriented classes and database schema. It can only represent some degree of semantics by adding keys or references. But RDF Schema is based on knowledge representation heritage. It takes ontology to represent the objects in the domain and the relationships between sets.
- RDF shows more demonstrability of knowledge than most other language.

Despite all these facts, RDF has limitations for developing full ontological description to model a complex system incrementally. It is often required to draw inferences about the compatibility of various combinations of components in complex system. Using RDF graph approach it is difficult to predict the graph of relationships among the components for assembling them in single modular construct.

RDF provides a basic vocabulary and grammar for representing assertions associated to any context, where any context is an environment within which some statement are held to be true and it applies to any particular circumstances. So, RDF schema which uses its own vocabulary needs to be further extended to establish a context from a global context by introducing external vocabulary.

For this reason this research aims to capture a higher level and modular construct of RDF which will be able to model a higher level of abstraction for components and to maintain a relationship among them. It is also essential to describe a flexible linkage between the value and the context, rather than depending only on the ontological relationship between the contents because ontological description can be revised and the value can be changed frequently. The approach to extend RDF to represent higher level constructs taking the above factor into account is described in Chapter 7.

6.4 Summary

This section highlights the issues that have been discussed in this chapter. Molecular biology database integration can use a variety of approaches, *e.g.* i. federated database with loosely coupled approach, ii. Federated database with tightly coupled approach and iii. Data warehousing. A clear distinction among these approaches are described and it is shown that how these approaches are related with the present work. Although database federation and data warehousing approaches are widely used techniques it is argued here that emergence of data in web or public domain has lead to the use of other approaches.

Developing a database federation requires to create a mediator for integration. A mediator represents the complete ontology of the subject domain for database integration. It is argued by different researchers that molecular biology database objects have different meanings in different contexts. It is also argued that it is not feasible to create an ontology for a subject domain, rather few overlapping ontology are required to represent a complete subject, specially, in the case of molecular biology database where data contents have different meanings. Thus, semantic conflict is a major challenge for creating such a database mediator and for describing an object description.

Another major challenge for a mediator is to generate a wrapper which will communicate between mediator and the data source to export the schema and to extract the data from the source. Unlike in business data, meaning in molecular biology data is not the same and data also do not reside in one place. This makes the task for wrapper generation a complex process. The wrappers are not also reusable for web based data. So the primary concern for web based integration is to find an efficient way to select a wrapper.

The data federation in the area of molecular biology are tightly coupled in order to avoid the complications. This implies that these databases are based on global schema for integration. It is argued that it is not possible to create the global schema for integration. Because global schema will not be able to serve the whole community in a particular subject domain.

It is proposed that publicly available data which are web based needs to be linked based on the context. In other words, the emphasis is on the meaning of the objects and how they relate to each other instead of just describing the object. In this way the object itself is not prioritised and the concentration is on the relationship of

the object and on defining the purpose of their relationship to each other. A collection of statements, a context, is described which is only true for a particular application domain and which is not true for the whole subject domain. In the process of implementing the context description other issues are revealed and it is apparent that these issues need to be resolved in metadata level. These issues are:

i. Need for navigational logic to locate the source from the web, and

ii. The need for an efficient language to describe the context and the object relationship

The main objective for any navigational plan is the ability to locate the appropriate source and the ability to execute the query on the source data which in this case is the web based molecular biology database.

The potential of using RDF as a medium of expressing context in metadata level is also analysed. The suitability and power of RDF in describing semantics of web objects are discussed. It is argued that the RDF in its present form is not sufficient to describe the application domain. The research has proposed an extension of the RDF to deal with this aspect.

The next chapter presents the proposed approach to implement the issues discussed in this chapter.

Chapter 7

Framework for Molecular Biology Resources Description and Navigation

Chapter Objective

Variance analysis of genetic data can be achieved by comparing data from different laboratory and public data resources. This chapter proposes a framework for interactions with different biological data resources based on the context of the web data so that a consolidated view of data can be achieved for variance analysis. A formal description of context of resources and their relationships have been described here using context graph. The context increases the interoperability by providing the description of the resources and the navigation plan for accessing the web based databases. A higher level construct is developed (has, provide and access) to implement the context in RDF for web interactions. The interaction among the resources is achieved by describing an integration domain based on the context. The integration domain allows to navigate and to execute the query plan within the resource databases.

Chapter Contents

Chapter 7

Framework for Molecular Biology Resources Description and Navigation

The real problem facing data integration is not the technologies involved, but in getting everyone to agree on the meaning of the term they use.

-Martin, 2001, Review article for Meeting on Trends in Biotechnology

7.1 Introduction

The chapter discusses a novel approach to search heterogeneous multiple biological resources. The implementation of a framework for resource description and navigation plan in Resource Document Framework (RDF) using context graph approach is proposed in this chapter. The framework initiates the searching for data set from multiple biological resources and it shows how the retrieved results from different biological resources can be integrated in a single page. This web based approach provides an alternative to the use of generic schema for database integration (Schonbach *et al.* 2000 and Markowitz, 1995b). The approach utilises loosely coupled schema (see chapter 6) which are less dependent on component data sources. It also coordinates the integration of component databases without depending on the top level schema/view model (Kemp *et al.* 2000a and Ram *et al.* 2002). This gives the scope to use laboratory based, target-specific component databases which are relatively independent and autonomous and which will be able to interact with other molecular biology data sources for more meaningful information without participating into any database federation (Kemp *et al.* 2000a). Most of the database integration researches are based on query optimisation or query script writing (Markovitz, 1995c, Kemp *et al.* 2000a, Kemper and Wiesner, 2001, Cheung *et al.* 2001 *etc.*). But the framework as proposed here describes how interactions with different biological data sources can be achieved with a single instance and without writing any query scripts. This utilises an integration domain to

extract the required information The proposed framework has the following unique features:

i. Increase the interoperability between the databases - the internal data model is always XML based even though the external data model could be text, HTML or Relational table based.

ii. Context description of the resources - these context are implemented in RDF which increases the interoperability by providing the description of the resources and the navigation plan for accessing the web based databases.

iii. Greater flexibility to design one's own navigational plan – this includes participating resources and query needs for a particular purpose.

iv. Image object keying - image content information is used for initiating the query and navigation.

7.2. Approach to Database Integration

Database integration approach is discussed in this section. The search scheme to obtain an integrated view of data sets which are distributed over different data resources, such as Protein Data Bank (PDB: D_p), Genome Data Bank (GDB: D_g), and Online Mandelian Inheritance in Man (OMIM: D_o) is described in Section 7.2.1.

The framework based on the search strategy is described in Section 7.2.2. Section 7.3 discusses aspects of meta data description and Section 7.3.1 proposes context graph model and a formalism for resource description. Section 7.4 describes the RDF model approach for resources description and Section 7.4.1 describes the context graph implementation in RDF model for resource description. An integration domain for navigational plan of molecular biology databases is described in Section 7.4.2 and 7.5. Section 7.6 presents an approach to describe the metadata for gel electrophoresis image spot. Finally an example implementing this approach is demonstrated in Section 7.7. The chapter ends with a brief summary and discussion.

7.2.1 Strategy for searching

Querying multiple heterogeneous molecular biology databases requires formulating a query based on the understanding of the comprehensive information of the molecular biology databases. For example, to find the *protein kinase* genes, the following steps are required: find the *protein kinase* products in the Genome Sequence Database (GSDB) *Product* class, find the genes associated with the same Feature as the *Product* and write the SQL scripts. It also requires to have the knowledge of accessionID for each entry. The *gdb_xref* attribute of *Gene* class in GSDB allows to find the appropriate *accessionID* attribute of Genome Database (GDB). This shows that one needs to have a high degree of knowledge in SQL script writing and in underlying structure of molecular biology databases to find *protein kinase* gene. This approach does not help the users to search any specific information easily. In view to this, this research has proposed an approach to make the integration and the query process easier. The concept is based on correlation. The correlation operation is carried out between the defective gene information from the dedicated component database and the normal gene information from the global databases. The approach allows the researchers to locate genetic map correlated to local defective gene data. The correlation operation is initiated by image object keying. The protein product corresponding to the image object keying is then searched from the global protein database (eg. Protein Data Bank, PDB). The relevant gene product disease information from Online Mendelian Inheritance in Man (OMIM) is then retrieved on the basis of this matching. It also retrieves the gene map from GDB, protein structural details and the corresponding image from the local database. The block diagram of this approach is shown in Figure 7.1.

This type of complex search needs to combine genetic, developmental, image, and other textual format of data. A context graph model (Section 7.3.1) is used to combine these multi-resources into the integration domain. The context graph represents the correlation among the public domain databases. A resource framework that supports model-based and hyper linked text data organisation can significantly increase searching capabilities.

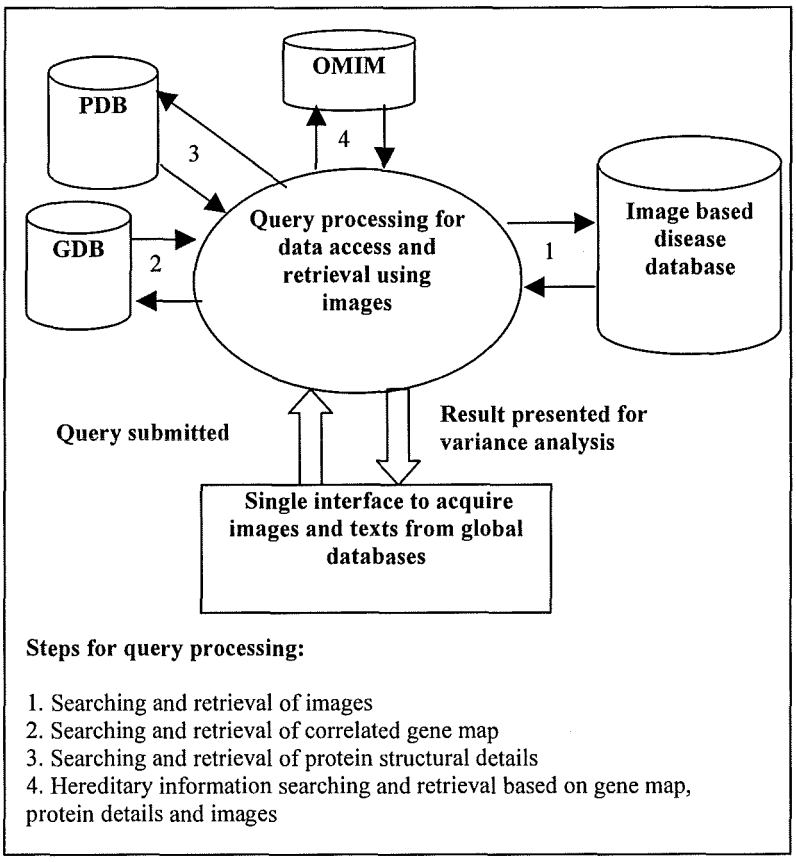


Figure 7.1 Query processing in cooperative environment.

7.2.2 A cooperative framework for database integration

An independent and dedicated database will store all the relevant laboratory results and images (Chapter 3, Section 3.1). The conceptual level of this component database consists of resource description and an integration domain. In addition, the component database also describes the mapping functions for resource linking. The database uses five converter modules/agents which are the integral parts of this framework model. A top level architecture of the framework is shown in Figure 7.2. These modules are mapping linker, meta data extractor, image feature extractor, dispatcher and result interpreter.

The search scheme employs these converter modules to scan the resource and navigational framework in order to retrieve the required information from the biological resources. The function of these modules are explained here (Khan *et al.*, 2002b):

i. *Mapping linker*- collects the resource mapping information from the component database. Researchers' will be able to configure their own choice of resource databases mapping information according to their specific query. For example, the retrieved information about any particular disease, the GDB, PDB and OMIM needs to be added in integration domain. This information will be passed to meta data extractor.

ii. *Meta data extractor*- collects meta data which represents the format used in structural details of the resource databases.

iii. *Image feature extractor* – collects image data features from the component database which are based on their contents. This will be used for image comparison and to understand the structural variances.

iv. *Dispatcher* – provides content based image descriptions from *Image feature extractor* and meta data from *Meta data extractor. Dispatcher* then submits the operators to the individual resource databases to establish the link.

v. *Result interpreter* - captures all the results from individual resource databases. It then presents the result in an integrated form to the user along with images and other related local information stored in the component database.

7.3 Meta data description for resources

Many researchers (Cheung *et al.* 2000, Kemp *et al.*, 2000, Markovitz *et al.*, 1996) proposed meta data based integration for molecular biology databases. Meta data were used to store the database structural description for schema comparison. This approach is widely used for schema transformation and to create datawarehouse, *e.g.* OPM model developed by Markovitz *et al.* (1996). Datawarehousing approach in molecular biology database integration has drawbacks, such as, information capacity in schema conversion, data update regularity, defining global schema *etc.* (Chapter 2).

Martin (2001) in a review article, "Trends in Biotechnology", emphasised the need for web semantics. He proposed the use of Resource Document Framework (RDF) for resource description. Martin termed RDF based resource description, or using robotics or agents, as '3rd Generation: semantic web' and suggested that it needs to be

employed for browser and application. The proposed framework attempts to utilise Resource Document Framework for resource description. However, the RDF model

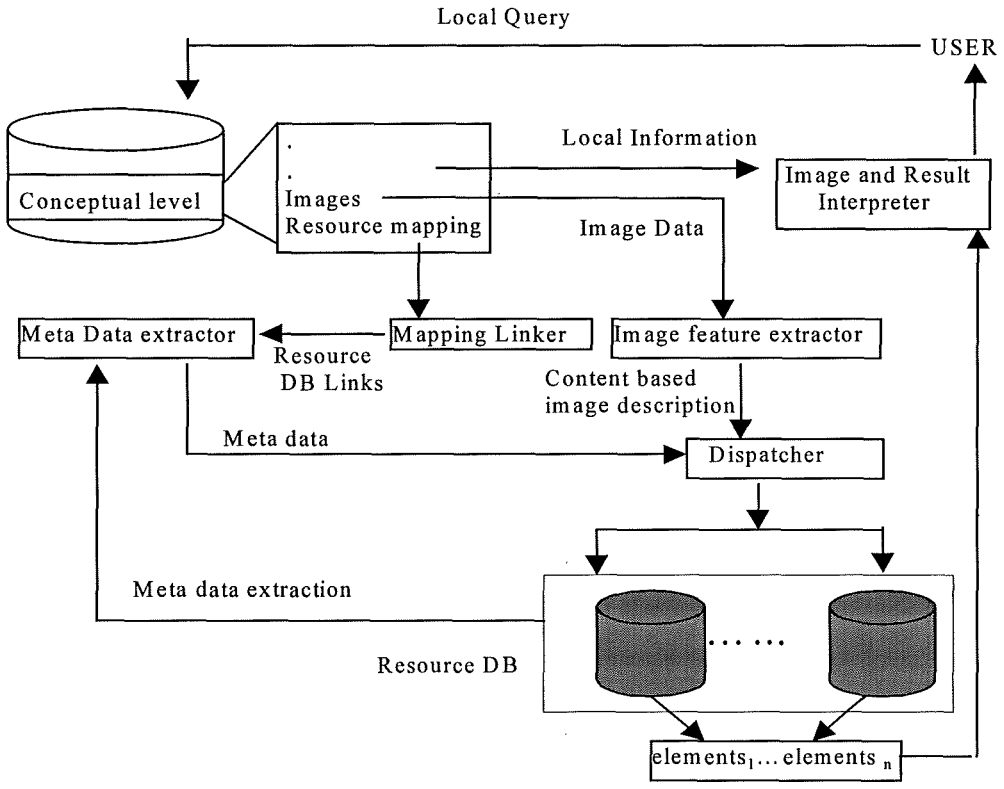


Figure 7.2 A top level framework architecture for multidatabase integration

needs to be extended further to higher-level constructs to tackle the complexity of the biological web. Because, the same data may exist at different web sites under different context or it may not follow the traditional approach for relational algebraic operation. For example, in GDB the gene *APP1* appears with structural information, *eg.*, gene location and gene annotation, but in OMIM it is only associated with disease information, *eg.,* Alzheimer's disease. This will not allow to correlate with each other using relational algebraic operation, such as union, intersection *etc.* The higher level constructs can include the context of the web resources which will increase the semantics of the webs. For example, context of *'genes responsible for diseases'* can

include all the associated resources and a derived correlation in the resource framework to increase the semantics of web for the integration purpose.

The following sections describe the context graph of resources to demonstrate the contents and links among the webs. The linking of the webs is based on the semantics and on the context of the purpose of integration. This research has proposed an approach showing how context of data item for a particular web can be included in the RDF for resource description and mapping. The proposed context graph model is described in the next section.

7.3.1 Context graph for resource mapping

The context graph model shows the links and relationships which exist between the public domain databases (Freidman *et al.* 1999). The context graph G is described using the following parameters:

- node name and unique ID
- operators: a set of values
- entry point in pages
- resource relation
- edges: links between the nodes
- labeled image object contents.

A graph G is defined with three interrelated subsets as: $G=(S, E, L)$, where S denotes the object in resource, *i.e.* page or any particular content, E defines the edges and L defines the element of an image object. S can be expressed as $\{u_1(v_1)...u_n(v_n)\}$ where u denotes the resource name or ID and v denotes the operator. $u(v)$ denotes an object v of a particular resource u. If any node is linked with another node, then it can be expressed with edge E as $E \subseteq u \times u$; where each $e \in E$ is represented as $e= u1.u2$ if e is edge linking resource u_1 and u_2. L denotes a labelled image object element with a list of values. The context graph also describes the entry point to nodes. In order for an entry point node to be accessed directly the operator needs to have a constant value. Figure 7.3 shows the context graph model for PDB, GDB and OMIM. This context graph model describes how the PDB, GDB and OMIM objects are related to each other and how they provide

access to other target pages either by means of node name (direct linking) or by using search forms. The diagram shows only those portions of the schema which are related to the integration objective or which share a common view for integration purpose.

Assuming any page *p(x)* and its relationship to other page *q(y)* so that:

- outgoing edges from page for different node exist in different server
- search forms on the page for an access to the target page in different node
- page leads to search form in different node for accessing the target page

To depict these scenarios the context graph is described as follows:

1. If a link from page *p(x)* to *q(y)* is labeled with an identifier *i*, this is expressed as

$$i(x,y) \xrightarrow{u_i \to u_j} q(y).$$

2. If a value *y* must be provided before accessing the target page, this is expressed as

$$i(x,y) \xrightarrow{form\ u_i \to u_j} q(y)$$

3. If a page leads to the target page form to provide a value *y* before accessing the target page, this is expressed as

$$i(x,y) \xrightarrow{u_i \to u_j} f(y) \xrightarrow{form\ u_j \to u_k} q(y)$$

Other parameters to describe a context graph for the resource webs are Page contents and Element relationship, Entry point relationship, Outgoing edges and Node type. The parameters with their meaning are described here and their representations are shown in Figure 7.3.

Page contents and element relationship: The node contents in the graphs are represented with thin one headed arrow (regular arrow) pointing from an object node *Node(u1)* to a *Node(u2)* which represents a property of *u1*. It also indicates the value relationship departing from a node and ending at a value node. For example, *Structure_biology* and *Geometry* of protein in PDB are described in the structure page and the *Geometry* information then leads to the *sturucture_conformation* page. Another example of node in the diagram is the *Type* and *Marker* in GDB shown in the *MAP* page.

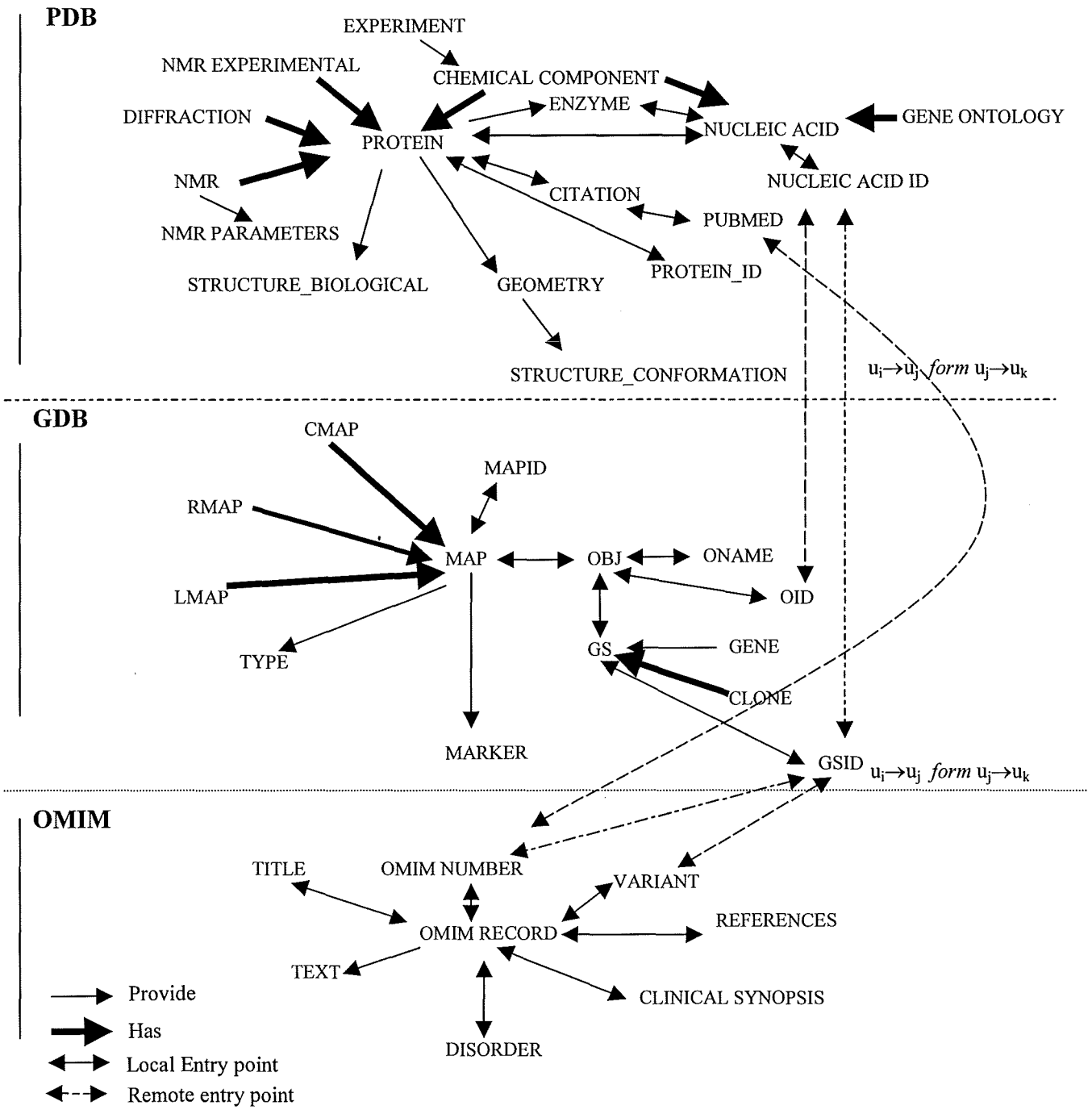


Figure 7.3: Context graph for the web contents and links

Entry point relationship: Entry point relationship is represented by an edge linking nodes u_i and u_j where u_i, $u_j \in S$. Double headed arrow in the graph describes the entry point of a page. The arrow linking for example, *nucleicacid_ID, protein_ID and PUBMED* provides the link to the structure of the proteins. In GDB, *GSID*, *OID*, *ONAME* and object *(OBJ)* provide the link for the gene map. Also, *GSID* provides the link for *GS* (gene sequence) and ultimately to the *OBJ*. *OBJ* then provides access to the gene map.

The elements disorder, clinical synopsis, OMIM number and gene variants in OMIM provide the link to the OMIM record.

Outgoing edges: outgoing edges from any page to the target page are described with dashed arrow head. One value node of one page will provide the target value node of another page. In this case, a search form, like links, maps relationship to other pages. The value of the target page parameter, *Y,* must be provided before accessing the link. This is expressed as follows:

$$u_i \rightarrow u_j \; form \; u_j \rightarrow u_k$$
$$i(x,y) \rightarrow f(y) \rightarrow q(y)$$

Because, node *i(x,y)* takes to the search form which leads to the target page with the given parameter y. It can be seen from Figure 7.3 that *Nucleic_acid_id* from page Nucleic Acid will provide a search form for GDB. The parameter *GDB_ID* will lead to the target page of GS (gene sequence). Consecutively, *GSID* of GDB web schema will provide a search form for OMIM entry. OMIM entry number or any variants of GSID will lead to the OMIM record.

Node type: Node types are used to describe the node elements such as Structure in PDB. For example, the contents of PDB are *Diffraction, NMR and Chemical components.* These contents make an element of a node which is *Protein* in our example. In GDB, CMAP (contig map), LMAP(linkage map) and RMAP (radiation hybrid map) make an element which is MAP. These node types are represented with thick arrow pointing towards an element of a page departing from the content.

7.3.2 Context graph interpretation for resource mapping

Each interpretation in context graph *I* defines a mapping *M*. A set of values for a particular mapping *M* is assumed to have a set of values called *V*. The map requires to contain triple '*has h*', '*provide p*' and '*access a*'. A simple interpretation *I* for mapping *M* is defined as follows:

1. A nonempty set *R* of resources, called the domain of *I* and superset of value *V*.
2. Resource *access a* points to the set of resources, *R1, R2..Rn*, if any value *x* is in *R1, R2..Rn* where *I(x)* identifies arguments for which the resources are true.
3. A resource R is composed of a set of elements *h* where $h=\{e_1, e_2, \ldots e_n\}$.
4. A resource provides a set of values p where *I(p)= true*.

To illustrate the mapping, from Figure 7.4, the mapping is for the *resource {r1:h, r1:p, r1:a}* where *<{r1:h, r1:p, r1:a}>= true* if any value *x* in *r1* for which interpretation *I(x)* is in resource *{r2:h, r2:p, r2:a}* and *<{r2:h, r2:p, r2a}>=true* if and only if any value *y* in *r2* for which interpretation *I(y)* is in resource *{rn:h, rn:p, rn:a}* and *<{rn:h, rn:p, rn:a }>*=true. In such case the map denotes all the objects in the resource.

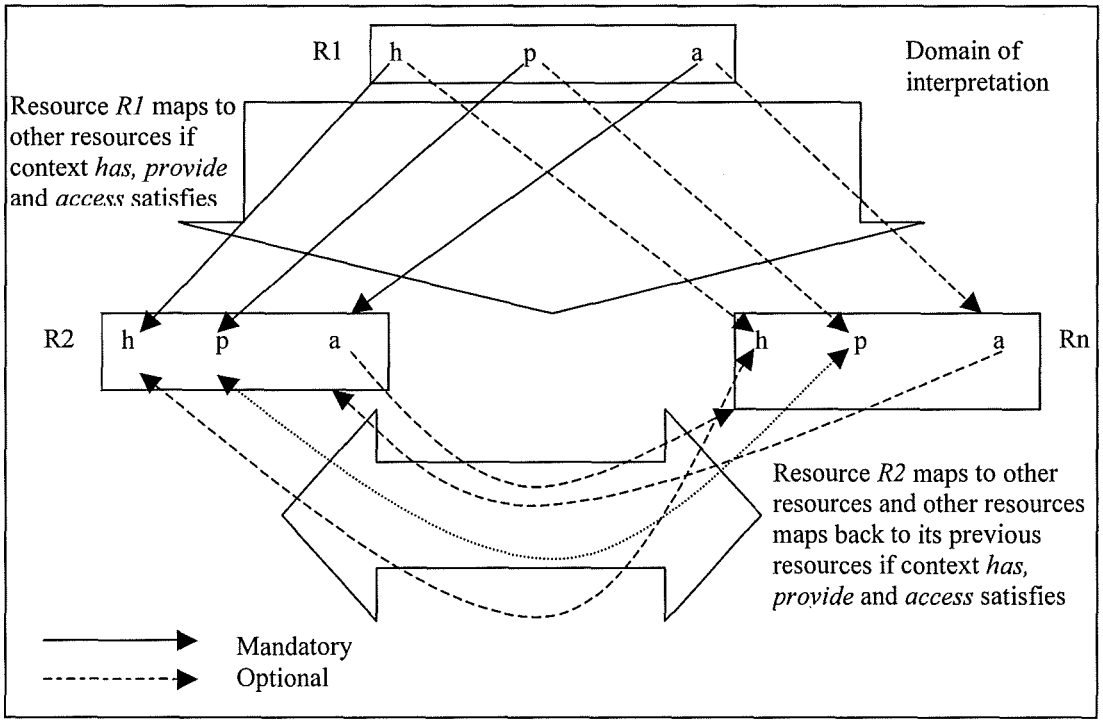


Figure 7.4. Resource mapping.

Any resource R_1 in Figure 7.4 maps to other resources if any of the context among *has h, provide p* and *access a* satisfies the other resources R_2 to R_n. R_2 and R_n also maps each other if the context value satisfies the resources. The resource from where the searching has started can point to one or multiple resources but it must point to at least one resource. The mandatory and optional resource mapping are denoted with solid and dashed arrow respectively in Figure 7.4. Because the dashed line is optional, so it is not mandatory that the accessed resource will map to other resources. If resource presents in any particular integration domain exceeds more than two, then the third resource needs to be mapped either by the starting resource which maps other resources or by the second resource which is accessed through the first resource.

7.4 Source Description with RDF

The main feature of RDF is to provide interoperability by adding semantics to the web resources. RDF describes the whole web page or part of the web page as resource and these resources are named by URI (Uniform Resource Identifier). Resource URI with their properties and values are used to define a RDF statement (Appendix C). The individual components of these URIs are described using the approach described by Berendt and Spiliopoulou (2000) and El-Beltagy *et al.* (2001). The summary of these components is presented in Figure 7.5. These URIs have the following meaning: the leading *hq* denotes protocol used to connect with other nodes, <hostDNS> is the DNS name of the host, where the searching is executed and <linktoPage> is the link to the target page. The *hq, <hostDNS>* and *<linktoPage>* are reffered to as URI prefix. The optional global parameter list and the object specific parameters are defined as "GlobRequest" and "ObRequest" respectively. These are used to provide the key values for the target page. The former one parameterised the remote element which refers to a set of elements or defines any global operation such as *search = Term & field = title* operation for OMIM and *explore* for PDB. The later represents any particular key values to reach the target page.

<webschema> ::="<hq>://<hostDNS>/<linktoPage>/[<GlobRequest>]?[<ObRequest>]"
<GlobRequest>::= <GlobRequest>=<GlobVal> "&" <GlobRequest>=<GlobVal>
<ObRequest>::= <ObRequest>=<ObVal>

Figure 7.5. Resource description syntax

RDF contains different URI prefixes and parameters for different resources which form an integration domain. The integration domain *D* in RDF has a set of triple which are $<Pi, \{SD_i\}, \{O_k\}>$. *Pi* denotes the attribute tag to access the URI prefix, *SD* denotes a set of URI prefixes and *O* denotes a set of key values for target page. For example, the PDB entries can be described in terms of attribute tag, source description and values using RDF modelling (Figure 7.6 and 7.7). RDF modelling uses *node* and *arc* to represent the source description. If an individual protein is identified by their *unique_ID 1AAP,* then the protein and structural details of that particular protein can be accessed from resource http://www.rcsb.org/pdb/cgi/explore.cgi. This scenario is modeled using RDF modelling in Figure 7.6.

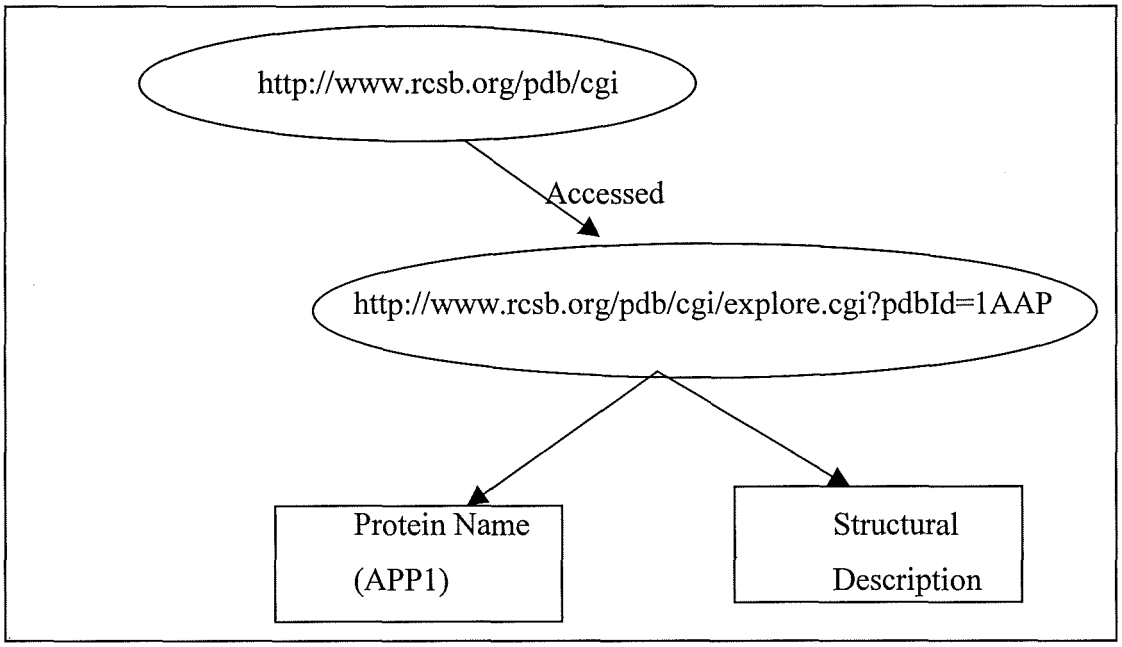


Figure: 7.6. RDF model for resource description

To represent a collection of resources, RDF uses an additional resource which identifies the specific collection. This resource must by declared to be an instance of one of the container object types. The *type* property is used to make this declaration. The membership relation between this container resource and the resources that belong in the collection is defined by a set of properties. These membership properties are named simply as "_1", "_2", *etc.*. Container resources may have other properties in addition to the membership properties and the *type* property, *e.g.*, additional statements. For example, a protein data bank resource contains individual protein descriptions which is modelled in Figure 7.7.

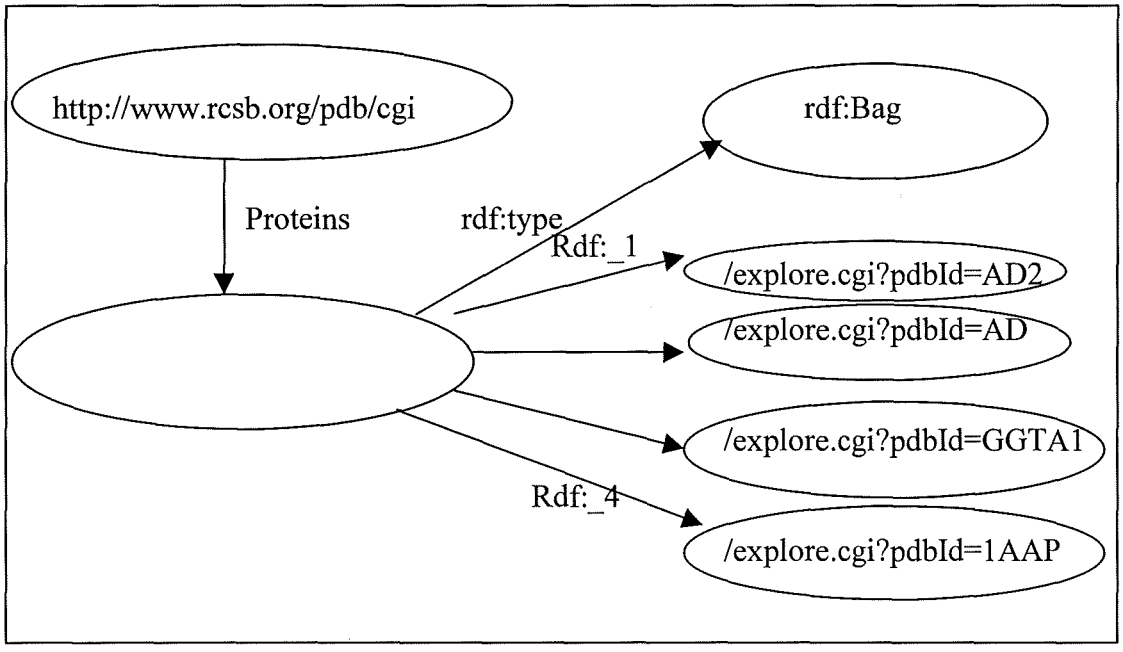


Figure: 7.7 RDF modelling for resource containers

A RDF/XML interpretation of the models as shown in Figure 7.6 and Figure 7.7 are described in Figure 7.8 and Figure 7.9 respectively.

```
<rdf:RDF
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:s=http://description.org/schmea/>
    <rdf:Description
     about=''http://www-lecb.ncifcrf.gov/cgi-
     bin/dbEngine/2dwgDB">
       <s:GelSpot>
         <rdf:Description
          about="http://www-lecb.ncifcrf.gov/cgi-
          bin/dbEngine/2dwgDB,getTableDataByID,WG00123''
          <rdf:type
          resource=http://description.org/schema/Proteins/>
          <v:ProteinID>       </v:ProteinID>
          <v:ProteinName>     </v:ProteinName>
        </rdf:Description>
     </s:GelSpot>
   </rdf:Description>
</rdf:RDF>
```

Figure: 7.8 Implementation of RDF modelling in XML

```
<rdf:RDF
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:s=''http://description.org/schmea/''>
   <rdf:Description
     about="http://www-lecb.ncifcrf.gov/cgi-
     bin/dbEngine/2dwgDB,getTableDataByID,WG00123''
     <s:protein>
        <rdf:Bag>
          <rdf:li    resource=''http://www.pdb.org'' />
          <rdf:li    resource=''http://www.gdb.org''/>
          <rdf:li    resource=''http://www.omim.og'' />
        </rdf:Bag>
     </s:protein>
   </rdf:Description>
</rdf:RDF>
```

Figure: 7.9 Implementation of RDF containers in XML

The importance of RDF is not that it is demonstrably better than any other form of knowledge representation, but it is widely used as an Internet standard, and it is designed to be used in an open web environment. Exchanging information between

computer systems or applications requires agreement about its representation. The RDF is meant to describe this agreement for web resources. But, for any complex system, such as integrating molecular biology databases, the basic RDF property-subject-object construct is not sufficient (Figure 7.6 and Figure 7.7) to describe the components of the resources without assembling the context relationship in RDF framework. The new factors, *has, provide* and *access,* as described in section 7.3.2 are included into RDF source description for high level abstraction of the participating components. By achieving context description of the participating components in RDF framework, high-level of interactions between the resource components can be captured which enable more detailed abstraction of the components. Another key feature of introducing 'context' into the RDF framework is that it describes the scenario where any component which is true for any specific resource is true again for any other resources. For example, describing the scenario 'genotypic and phenotypic details of Alzheimer's disease' requires several components to be implemented first, such as, phenotypic details of Alzheimer's disease, genotypic details of Alzheimer's disease, *etc.* Any particular resource can be mapped to component of the 'phenotypic details of Alzheimer's disease' scenario, *e.g.*, in PDB, but any other resource, *e.g.*, SWISS-Prot can also be mapped to the component for any other purpose, *i.e.* linking with 2D gel electrophoresis images. This level of complexity and user's choice can be accommodated in RDF framework

The next section describes how the context is implemented in RDF statement for component abstraction and navigation.

7.4.1 Context representation in RDF

RDF is a collection of statements. The statements in RDF explicitly define the associated resources and their properties. Thus RDF itself is the collection of resources. The resources in RDF can be referred by the properties of the RDF and these property values can be used to indicate different relationships between the resources.

The implementation of context in RDF is not direct as it is for resource description. The context description in RDF requires to have the following characteristics:

- Contexts may be related to other contexts in various ways.
- All the contexts have their own values, properties and resources
- The values of the contexts determine the contents of the resources
- The properties of the contexts determine the reasons of the resources
- The resource access provides the target for any particular elements.

These characteristics are expressed by describing a context as a statement set with some additional structural and logical properties. These are explained as follows:

- *rdfc:context* is a subclass of rdfc:StatementSet, and it represents a context. By inheritance this consists of a set of statements. Context implements a set of statements which provide values, properties and resources. A context can be a set of contexts for any particular purpose. For example, any context for resource PDB can be described as:

  [ProteinDataBank]--------rdf:type----→ [rdfc:Context]
  {
  statement of PDB values
  statement of PDB properties
  statement of other resources
  }

  In the above example, ProteinDataBank is a context for PDB resource. This ProteinDataBank context has its own values and properties. It also provides the URI for other resources.

- *rdftype:has* indicates values that is a member of a context, and which is also asserted to be true for a particular resource. This corresponds to the values that the resource has. For resource PDB it can be described as follows:

  [ProteinDataBank] --------rdf:type --→ [rdfc:Context]
  {
  [www.rcnbi.pdb.org] ------ has → "NMR"
  [www.rcnbi.pdb.org] ------ has → "DIFFRACTION"
  [www.rcnbi.pdb.org] ------ has → "CHEMICAL COMPONENTS"
  }

In this example, PDB resource has NMR, Diffraction and Chemical components in the context of ProteinDataBank.

- *rdftype:provide* indicates properties to show the particular reasons for accessing the resource. This is true for one particular resource in one context, but this can also be true for any other resources in another context.

[ProteinDataBank] --------rdf:type --→ [rdfc:Context]
{
[www.rcnbi.pdb.org] ------ has → "NMR"
[www.rcnbi.pdb.org] ------ has → "DIFFRACTION"
[www.rcnbi.pdb.org] ------ has → "CHEMICAL COMPONENTS"
}
{
[www.rcnbi.pdb.org] ------ provide → [BiologicalStructure]
[www.rcnbi.pdb.org] ------ provide → [Geometry]
[www.rcnbi.pdb.org] ------ provide → [Enzyme]
[www.rcnbi.pdb.org] ------ provide → [ProteinID]
[www.rcnbi.pdb.org] ------ provide → [NucleicAcidID]
[www.rcnbi.pdb.org] ------ provide → [PubMed]
}
{
BiologyStructure: [Strucutre] ----- has → ……..
.
.
.
ProteinID: [PID] ---- has → .....
NucleicAcidID: [NID] ---- has → ……....
}

In this example, the resource is described with a set of values within the context of *ProteinDataBank*. The context graph described in Figure 7.3 is implemented in this example. Within the context of *ProteinDataBank* the resource has a set of values, *NMR*, *Diffraction* and *Chemical components*. This resource provides Geometry, Enzyme and BiologyStructure of a protein. This BiologyStructure, Geometry and Ezyme have its own values which are implemented by *has*.

- *rdfctype:access* are the remote or local URI targets to have access to any particular element. For any resource PDB it can be described as follows:

[ProteinDataBank] --------rdf:type --→ [rdfc:Context]
{
[www.rcnbi.pdb.org] ------ has → "NMR"
[www.rcnbi.pdb.org] ------has → "DIFFRACTION"
[www.rcnbi.pdb.org] -----has → "CHEMICAL COMPONENTS"
}
{
[www.rcnbi.pdb.org] ------ provide → [BiologicalStructure]
[www.rcnbi.pdb.org] ------ provide → [Geometry]
[www.rcnbi.pdb.org] ------ provide → [Enzyme]
[www.rcnbi.pdb.org] ------ provide → [ProteinID]
[www.rcnbi.pdb.org] ------ provide → [NucleicAcidID]
[www.rcnbi.pdb.org] ------ provide → [PubMed]
}
{
BiologyStructure: [Strucutre] ----- has → ……..
.
.
ProteinID: [PID] ---- has → .....
NucleicAcidID: [NID] ---- has → ……....
}
{
[PID_1] ---------- access → www.rcnbi.pdb.org
[PID_2] ---------- access → www.omim.org.cgi.bin?=value
[NID_1] ---------- access → www.gdb.org.cgi.bin?=value
}

In this above example, *access* indicates the target page of the web within the context of ProteinDataBank which is true for a particular *PID (protein ID) and NID (nucleic acid ID)*. Any given values for *PID_1, PID_2* and *NID_1* will lead to the target page of resources. The context of GDB and OMIM resources are described using similar approaches.

7.4.2 Integration domain using context

A context is the collection of attributes of any resource where a resource is described in terms of its values, objects it is providing and any connection to other resources. The set of expressed contexts for each resource is integrated by creating a unifying context for all the contexts. This unifying context is used as the domain or

range of integration. It leads to all the resources which are associated with each other and which represent a collection of statements describing the objects present within it. This also allows any context to hold another context without knowing the detail physical structure. This provides a modular approach for describing any high-level relationships among the components. The following example shows how to integrate PDB, GDB and OMIM.

[IntegrationDomain] ---------rdf:type → [rdfc:type]
{
[ProteinDataBank] --- provide → [PID]
[ProteinDataBank] --- provide → [NID]
{
[PID] ----------has → "value"
[NID] ----------has → "value"
}
[GenomeDataBank] ---provide → [GID]
{
[GID] ---------has → "value"
}
[OmimRecord] ----provide → [OmimNumber]
{
[OmimNumber] ---- has → "value"
}
[ProteinDataBank] --- access → [GenomeDataBank]
[GenomeDataBank] --- access → [OmimRecord]
[ProteinDataBank] --- access → [OmimRecord]
}

The contexts, ProteinDataBank, GenomeDataBank and OmimRecord, are placed under the new context *[IntegrationDomain]* which is unifying all the contexts. The resources as a whole is attached to another resource, *[IntegrationDomain]* in this example. Any changes in the resource will not affect the integration domain. This initiates the query for the given data, such as *protein ID (PID),* gene *sequence ID (GID) and Omim record ID (OMIMNumber),* to look for the matching data value within their individual contexts. The Integration Domain provides a navigational plan to explore and execute the logical plans for a set of resource webs. In the above example, a map is described using link, *e.g.* ProteinDataBank provides link for GenomeDataBank, GenomeDataBank provides

link for OmimRecord and ProteinDataBank also provides link for OmimRecord. The values for PID, GID and OMIMNumber provide the operators for accessing the web nodes individually. Thus, basically, the Integration Domain is providing a navigational plan to explore and to execute the logical plans for a set of resource webs. The integration domain selects the locations which is associated with a resource and then a set of values, such as, PID, GID and OMIM number, act as relation atom to provide the link with the resources. For example, the web which has protein information needs to link with web which has gene information. The relation atom for these two web resources are Protein ID and Nucleic Acid ID. For instance, we can reach the GDB node with particular GSID if the PDB node with particular PID and NID are provided and if PID and NID act as relational atom. So, it can be expressed as

$$PDB(PID) \wedge PDB(NID) \overset{PID,\ NID}{\Rightarrow} GDB(GSID)$$

7.5 Search Initiation

The search initiation for target pages residing in multiple resources starts by taking the input streams from the integration domain (Section 7.4.2). The integration domain provides the maps for the resources and supplies a set of values to reach the target page. The search plan is based on the integration domain. The integration domain acts as a virtual table and the search initiator receives the input data from this table. It passes the data to the hyperlinks in order to reach the target. All specific search operators and maps of the hyperlinks for any individual integration plan are transmitted to the search initiator, the *Dispatcher* (Kemper and Wilsner, 2001). The *Dispatcher* then submits the operators to the individual resource databases to establish the link. Hyperlink for the search mechanism is carried out by the Dispatch operator (Kemper and Wilsner, 2001). The basic functions of the dispatcher are:

i. the search operators are allocated to its target hyperlink for specific target.

ii. checks the context graph described in RDF for multiple bioinformatics sources and for mapping more details.

iii. collects elements from each target to compose a whole object and to determine each map with search operator as a sub-plan of the total integration plan

iv. creates dynamic memory to hold the subset elements of the output for further integration into a whole object

A proposed search initiation mechanism is described in Figure 7.10. It is a bottom up approach to receive the elements from each node and then to pass through to the next node to receive more elements from the nodes. Finally, when all the elements are collected from the target nodes, it is then combined into one whole object. For instance, if data set d is distributed over a number of biological resources, D_p, D_g *and* D_o, then to retrieve d, the following steps are performed:

i. access to the D_p, D_g and D_o resources;

ii. retrieve the required pages p_p, p_g, p_o from D_p, D_g *and* D_o;

iii. access the required set of elements e_p, e_g and e_o from the pages p_p, p_g, p_o respectively and then pass the elements of the pages to the result integrator to embed the elements into a single page p. These steps are shown in Figure 7.11.

The overall objective of the dispatcher is to apply a set of search operators O to the respective data resource R as described in mapping linker and let $O_i(R_i)$ $(1=<i=<n$, where n is a finite value) be a set of derived facts related to the overall search result. A constructor module based on this concept is implemented to extract the element contents from the target HTML pages. The basic construct of the module is shown in Figure 7.12.

An integrator module is created to integrate the captured results from individual resource databases. It presents the results in an integrated form to the user along with images and other related local information that are stored in the component databases (see Chapter 3). Document Object Model (DOM, 2000) has been used which is an Application Programming Interface (API) used for HTML and XML documents (Appendix C). It defines the logical structure of the documents. A converter module is then used to convert the DOM file into XML documents for interoperability. A segment of the converter module is shown in Figure 7.13.

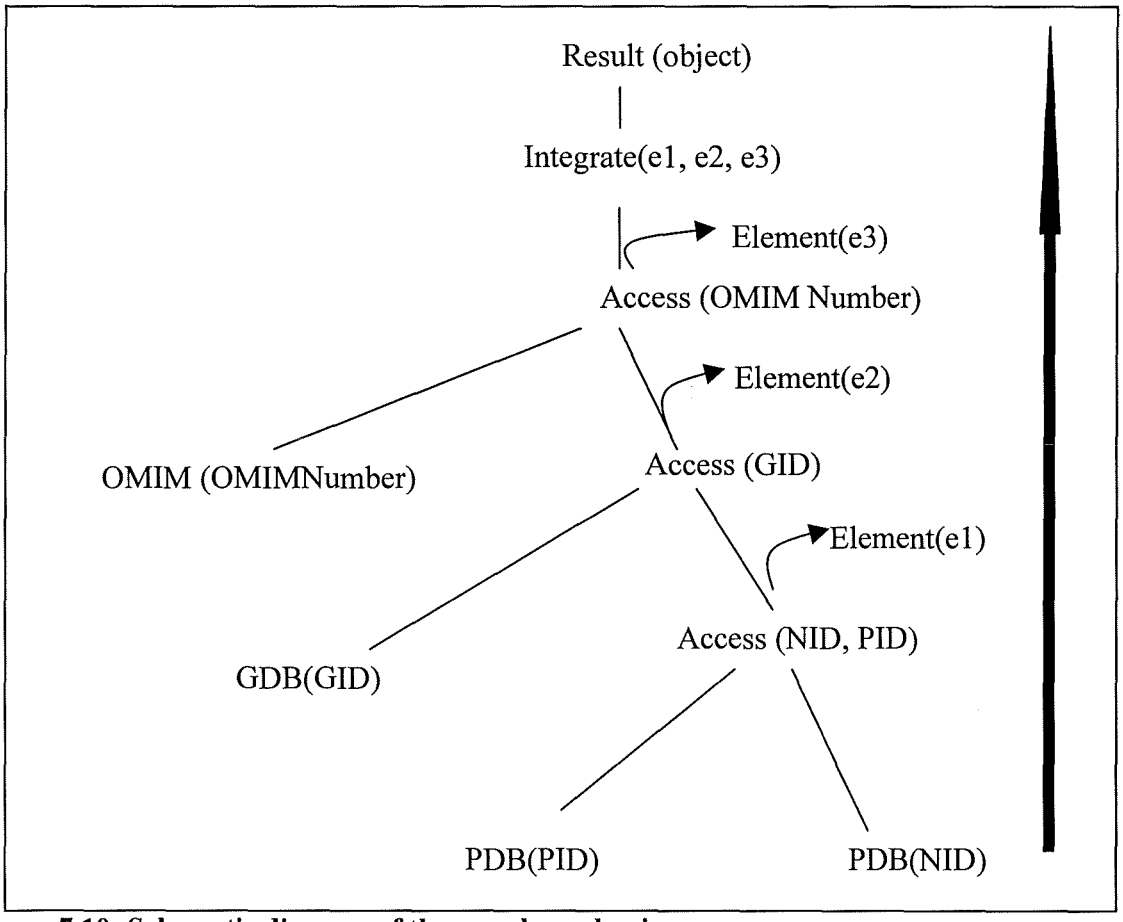


Figure 7.10: Schematic diagram of the search mechanism.

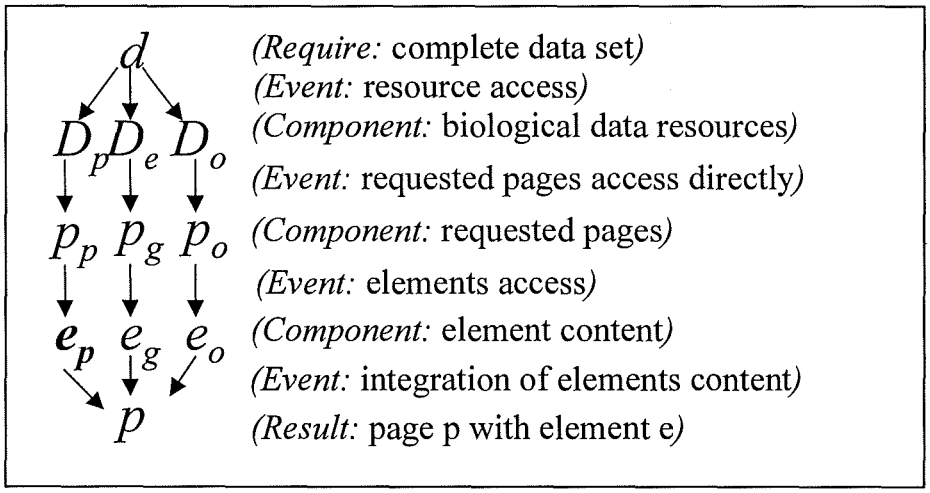


Figure 7.11: Events and Components for searching the element content

```
public static String getText(String URIStr) {
        final StringBuffer buf = new StringBuffer();
        try {
        HTMLDocument doc = new HTMLDocument() {
               public HTMLEditorKit.ParserCallback getReader(int loc) {
                    return new HTMLEditorKit.ParserCallback() {
                public void handleText(char[] data, int loc) {
                            buf.append(data);
                            buf.append('\n');}};}};
            URL url = new URI(URIStr).toURL();
            URLConnection conn = url.openConnection();
            Reader rd = new InputStreamReader(conn.getInputStream());
            EditorKit kit = new HTMLEditorKit();
            kit.read(rd, doc, 0);
        } catch (MalformedURLException e) {
        } catch (URISyntaxException e) {
        } catch (BadLocationException e) {
        } catch (IOException e) {
        } return buf.toString(); }
```

Figure7.12. Constructor module to extract the element contents from the HTML

```
 try
{source = new DOMSource(doc);
 File file = new File(filename);
 Result result = new StreamResult(file);
Transformer xformer=
TransformerFactory.newInstance().newTransformer();
 xformer.transform(source, result);
 } catch (TransformerConfigurationException e) {
  } catch (TransformerException e) { }  }
```

Figure: 7.13 Converter module to transform DOM documents into XML

Finally, an Extensible Stylesheet Language for Transformation (XSLT) (Wadler, 2000) template is used to extract the XML elements from the converted XML files.

7.6 Image Feature extractor

This module collects image data features based on their contents from the component database (Chapter 5). This is used for image comparison and to understand the structural variances. An image content extractor module (M_f) is developed to match protein spots between source and target gel electrophoresis images. The module identifies the protein spot in the target image which lies on the same line of path as it is in the source image (electrophoretic mobility concept). A shape matching algorithm using Generalised Hough Transform and Canny Edge Detection method have been used

to determine the shape variance. The set of spots in each gel electrophoresis is labelled and each spot is associated with a finite number of values, *e.g.*, accession number of gene, protein and OMIM in addition to protein spot shape, mean value and positional vector. A class is formed using these labels and values. Each entry in the class is of the form (L, V) where L is a label and $V=\{v_1, \ldots, v_n\}$ is a set of values. Each v_i represents a value that could potentially be assigned to an element E, if label (E) matches L. In our case element E is a spot which corresponds to the specific 3-D structure of a protein. M_f, is used to look for the value v_i in order to begin the search for the corresponding element E in the target image (Khan *et al.* 2003 and Khan and Rahman, 2003).

7.7 An Example of Integrating Biological Resources

Integration approach is implemented by using the tools as described in the above sections. DOM interface is used which provides a set of API calls for accessing the content of the documents; a special wrapper designed for DOM-compliant data sources exports this information to the dynamic buffer (a virtual table) for each such API call. The input parameters for the DOM calls are *accession numbers* of the biological resources which are obtained by scanning through the RDF data. To illustrate the process, an example to find information on Alzheimer disease is shown here. The protein spot (*APP1*) for Alzheimer's disease is selected from the gel electrophoresis image (Figure 7.15a). The spot position is then matched in the target image and the 3D structural protein image for *APP1* is retrieved by matching the image content elements using the image content extractor (Figure 7.15a). The *Dispatcher* now dispatches the searching operators individually to the respective data resources (Figure 7.14) for unifying them into a single page. For example, the following individual resources along with their operators (collected from RDF) are sent to the respective databases for information on Alzheimer's disease and to correlate the gel protein spot with the *geno* and *phenotypic* information. DOM interfaces are used for the following resources and operators to traverse along the contents.

Resource r1← http://us.expasy.org/cgi-bin/niceprot?Operator s1.PI=P05067

Resource r2← http://us.expasy.org/cgi-bin/blast.PI? Operator s2 sequence=P05067

Resource r3← http://www.rcsb.org/pdb/cgi/explore.cgi?Operator s3 pdbId=1AAP

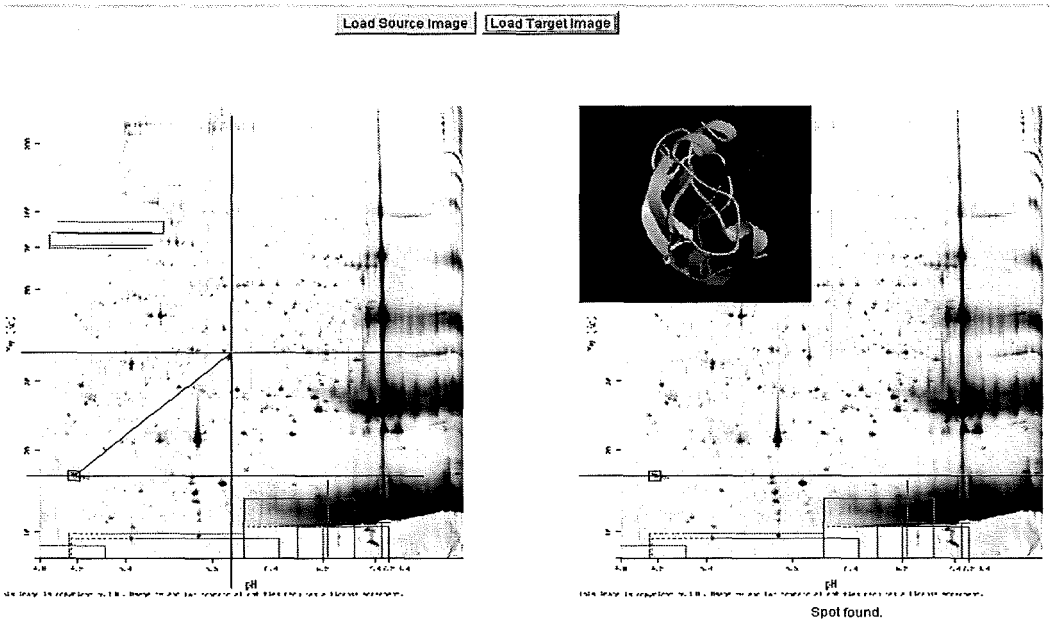
Resource r4← http://www.gdb.org/gdb-bin/genera/accno?Operator s4 accessionNum=GDB:119692

Resource r5← http://www.ncbi.nlm.nih.gov/htbin-post/Omim?Operator s5 dispmim=104300

Figure. 7.14. Node operators to be dispatched for target node.

Figure 7.15(a) shows the interface where a source gel electrophoresis image is loaded and the spot for *APP1* protein is selected. A target image is searched until a match is found. It is then loaded with spots in the same position. The contents of the spots are matched with RDF document which enable to find the resources and operators in order to search for other details, for example, Figure 7.15(b) is the single page of HTML which consists of the data elements retrieved from multiple databases. These data elements are retrieved and unified using the *Dispatcher* and *the Result Integrator*.

Figure 7.15(b) shows the final combined result in HTML which has the elements from different biological resources, *e.g.*. mutation rate, likelihood in male and female, phenotypic details, diagnosis environment, pathological lesion details and coding region. All the elements which are collected from different resources are embedded in a single page called 'Trait Analysis'. The elements which are collected from different resources are given in Table 7.1.



a

Trait Analysis
Mutation Rate: Peak lod score with Gm 1.37
Likelihood in male and female: Theta =0.05

Phynotypic Details: Protease Inhibitor Domain Of Alzheimer'S Amyloid beta-Protein Precursor (APPI) - Chain A 1AAP:A : VREVCSEQAE TGPCRAMISR WYFDVTEGKC APFFYGGCGG NRNNFDTEEY GGGGS SSS EEE EEEETTTTEE EEEEE SSS S BSSHHH CMAVCGSA HHHHH Protease Inhibitor Domain Of Alzheimer'S Amyloid beta-Protein Precursor (APPI) - Chain B 1AAP:B: VREVCSEQAE TGPCRAMISR WYFDVTEGKC APFFYGGCGG NRNNFDTEEY GGGGS SS EEE EEEETTTTEE EEEEE SSS S BSSHHH CMAVCGSA HHHHT
Diagnosis Environment:
CSF analysis of hyperphosphorylated tau protein. (Phosphorylation at serine 199 tau-199) for the antemortem diagnosis of AD
Genetic Event Type: Mutation in APP (A-beta amyloid precursor protein gene). Autosomal dominat trait in families
Abnormality Frequency:7 of 21 families. 3 generation transmission, Familial incident 43%
Sex: Male
Ethnicity: Afro-British
Region: South Africa
No. of case Studies:001

Pathological lesion
Details: Alzheimer's disease/cerebral hamorrhage, Dimential, Trisomy 21, Parkinsonism, long tract signs. Overall Mean onset Age 45.7 and overall mean age of death 53.3. 20% cases
Coding Region: Point mutation; From GCA to GGA. Observed in g.275267>g and p.A692G. Region is Ex17 and N-term

b

Figure 7.15: (a) APP1 protein spot matching and (b) Element collection and integration.

Table 7.1: Elements Collected from the Resources

Elements	Resources
Mutation rate	Online Mendalian Inheritance in Men
Likelihood in male and female	Online Mendalian Inheritance in Men
Phenotytypic details	Protein Data Bank
Diagnosis Environment	Online Mendalian Inheritance in Men
Pathological lesion	Online Mendalian Inheritance in Men
Coding Region	Genome Data Bank
Genetic Event Type	Genome Data Bank
Sex	Local database
Ethnicity	Local database
No. of cases	Local database

7.8 Summary

In this chapter a solution to the problem of searching elements from different biological resources in heterogeneous environment is presented. The framework for the solution proposes a loosely coupled integration mechanism which is based on web data.

Section 7.3 outlined the overall strategy for searching the molecular biology databases. The search scheme uses public domain molecular biology databases, *e.g.*, PDB, GDB and OMIM. The scheme also employs a component database which is laboratory based. The component based database is dedicated, target specific and autonomous in nature. This component database initiates the links to resource databases using image object keying. The search scheme utilises multiple key values to access the target nodes of the resource pages.

A top level architectural view of the framework is proposed in Section 7.3.1. This architectural model integrates several agents, *e.g.*, metadata extractor, image content extractor, mapping linker, dispatcher and result interpreter. The cooperative collaboration among the agents leads to the integrated view of all the elements that are collected from different resources.

The successful technical implementation of the framework depends on the successful description of the molecular biology resources. Section 7.4 proposed a formalisation for resource description. In this section, it is argued that the use of context needs to be implemented with the metadata of resources. The meta data does not need to depend on the schema structure but it needs to depend on the context of the contents. The author proposes the following three new parameters: *'has'*, *'provide'* and *'access'* for formalising the context. A novel context graph approach to describe the web contents on the basis of the context (Section 7.4.1 and 7.4.2) is also presented. This approach attempts to use the semantics used within the resources instead of using the schema structure.

In particular, the context graph implementation is the basis of database integration as it provides the resource mapping, contents and an environment for navigational plan which is only true for a particular context.

Resource Document Framework (RDF) is chosen to describe the context for the resources. The major challenge to describe the context in RDF is that the RDF lacks higher-level constructs for describing the context. The author proposed a higher level modular construct for implementing the context into RDF in Section 7.5 and 7.5.1. This approach is extended further to construct an integration domain with participating resources and data values for navigational plan.

Section 7.6.1 described the search initiation mechanism. The search initiation for the biological resources take place in multi phases. A Dispatcher module starts the search initiation by fragmenting the plan and then by allocating them to the appropriate resources. For each step, a new set of elements are collected from the target pages and subsequently all these elements are integrated by the Result Interpreter which combines the elements into a single page.

Although the problems for integration of molecular biology resources are complex, the solution for an efficient method lies in describing the semantics of the webs. The author attempts to describe the semantics by using context in RDF and proposes a number of agents for intermediate conversions and interpretations.

The unique contribution of this approach is the Image Feature Extractor module which extracts the image contents from component databases and correlates it with the integration domain. The function of this module was discussed in Chapter 5 and in Section 7.6 of this chapter. Section 7.7 showed how this function can be included with the integration domain for search initiation using image object keying.

In conclusion, a novel approach for database integration is presented in this chapter. This approach is an alternative to any mediator based integration. The mediators rely on the central schema and they are responsible for answering queries within this schema. In other words, the query is local and schema is global in the mediator based integration. In contrast, an integration domain instead of the global schema is proposed in this approach which will provide the navigational plan and the participating resources for integration. This approach gives more flexibility for integration by including more resources and their context or by deleting any resources. An example to find the details on Alzheimer's disease is presented in Section 7.8 to demonstrate the framework.

Chapter 8

Discussion: Integration of Component Database Based on Image Object Selection

Chapter Objective

This chapter summarises all the issues that are presented in this research. It also discusses outcomes, limitations and possibilities to draw a concluding remark about this research. The chapter compares the proposed framework with other research to establish the novelty and originality of this research. Finally the chapter sets out propositions for future work to be carried out in this area.

Chapter Contents

Chapter 8

Discussion: Integration of Component Database Based on Image Object Selection

8.1 Introduction

In the preceding chapters, the thesis has examined a wide range of issues in the area of database interoperability and the use of image object selection for databases linking. This chapter summarises the thesis in the context of the issues and methodologies raised in this research and outlines the novelty of the proposed approach for database integration in the area of molecular biology. The thesis also reflects on how far the core issues of database interoperability and the development of a framework for cooperative environment have been addressed. The research makes suggestions for further work in the area of molecular biology database interoperability.

Section 8.2 summarises the motivation and research issues that have been identified from literature review in Chapter 1 and Chapter 2. The section highlights the concerns for issues which have not been addressed yet. Section 8.2.1 describes the approaches taken to address these research issues and it shows how to correlate these approaches with the present practices in this research area. Section 8.2.2 emphasises the importance of establishing theory to improve database interoperability without affecting the semantics of data, database architecture and local autonomy for future practice. Section 8.3 concludes with suggestions for further works in this research area. Section 8.4 presents a concluding remark about the research.

8.2 Summary and Discussion of the Thesis

Bioinformatics research is currently facing a major challenge to integrate the genome databases (Robbins, 1996; Markowitz and Topaloglou, 2001; Donnelly, 2003 and Gieger *et al.,* 2003). This research has investigated the format and structure of the

current molecular biology databases. In the process of this investigation, the research has outlined a clear understanding of the molecular biology database structures, their functions and their functional importance in the area of genomic data analysis.

The research has also examined the software and database management systems which are emerging in the area of bioinformatics. The uses of these utility software have been examined in the thesis and a correlation has been drawn between software and software users. The thesis has focused on independent laboratory users and genomic data analysis tools for variance analysis of gene mutation data. The research has established the need for the following theoretical aspects: designing data model for gene mutation data, determining a framework for cooperative environment to share the knowledge accessed from different data resources, dealing with database interoperability and outlining the theoretical approach for using image matching in gel electrophoresis images for query initiation.

Chapter two has set the scene for this research. The chapter has reviewed the issues of molecular biology database 'interoperability' and it has described how heterogeneous nature of the molecular biology databases is hindering the integration process with each other for sharing the data and the knowledge. On the basis of literature review and empirical analysis of current databases, their data and schema diversity have been examined to understand the reasons for impediments of database interoperability. In this regard, Liu *et al.* (2000) commented about the complexity of designing large databases and outlined how components derived from multiple-data sources are increasing the density level of databases. The chapter has also presented the drawbacks of the current approaches and this initiated the scopes for this research. The research has raised the following issues on the basis of the literature review:

- Integration molecular biology databases for gene variance analysis and keeping all the technological aspects transparent to the users.
- Preserving local autonomy of databases for data submission while maintained and created by local authority.
- Initiating query without writing any type of script languages and submitting query to the resource databases.
- Exploring data dynamically using automated navigation across the resource databases.

Chapter three has addressed these research issues that have been raised in chapter two. The chapter has discussed the methods and approaches adopted for this research. The proposed approach addresses the issues for developing a component database for database integration. To overcome the issues raised by Liu *et al.* (2000), the research has proposed to implement a component database which is independent of any participating component database. This component database will not have any subcomponent of any super component database nor it will have any super component derived from any participating component database to integrate the component databases with federated information system. The research has employed a number of agents to facilitate parsing queries based on the image object selections (Chapter 5 and 7). The research has proposed the application of image object keying as an alternative to query script writing for multidatabase query.

The following section summarises the theories and the concepts which underpin the implementation of the proposed framework.

8.2.1 Developing the theoretical concepts

The main goal of this research is to establish the concept to deal with the research issues that have been raised in this research. In addressing these research issues, the preceding chapters have gone a considerable way to discuss them in depth. Chapter two has set the background for research issues and chapter three has developed the theoretical concept to address the issues.

To establish the theoretical concepts, the research has reviewed literatures for issues on database interoperability and it has investigated terms like 'collaborative database management' and 'federated database management'. Although the review has found a number of approaches for database integration, but none of these approaches has actually addressed the criteria described by Markowitz and Ritter (1995) and Letovsky (1995). 'Federated database' and 'data warehousing' approach are two popular approaches among these. The research has established why an alternative approach to address the data integration problem is necessary and it has shown how it can be achieved by using a dedicated, target specific and independent component database. It has suggested a framework for web based multidatabase query and it has shown that how it can be initiated from the component database.

8.2.2 Comparing this research with other research

Kemp *et al.* (2000a) introduced a mediator based approach to integrate genome databases which is based on Wiederhold's (1992) proposal. In their technique they have used metadata to describe the integration schema and the external schemas of each of the federation's data resources. This approach depends heavily on the schema, which leads towards the 'global schema integration' approach. The limitations of this approach has been discussed in chapter two. Another approach is TAMBIS (Paton *et al.*, 1999), where they maintain a classification hierarchy of protein with many pre-defined subclasses. This technique works well when the query meets the fixed framework of the hierarchy and returns the values of corresponding member to the specific subclasses. This approach works only when the query is submitted with in a specific framework, *i.e.*, it should match with the local schema classes. However, unlike 'TAMBIS' approach, this research has described a non-redundant schema integration concept. In this approach the local schema will not hold any subcomponent, union or intersection of other objects of resource databases. This concept will not allow any redundancy of attributes and thus it will be less prone to loose data integrity. It also reduces the risk to loose information capacity.

Both Kemp *et al.* (2000b) and Paton *et al.* (2000) used 'wrapper' in their 'mediator based integration' and in TAMBIS respectively. Kemp *et al.* (2000b) used 'wrapper' to generate code in a different query language, whereas Paton *et al* (2000) used the term 'wrapper' as a black box interface to remote resources. But, this research has attempted to develop intermediate database engines for metadata extraction from resource database which is dynamic and not static. A mapping linker will provide the necessary information to the metadata extractor to map to the resource databases. This transient metadata information which is collected by metadata extractor will be fetched into query generator for generating the necessary code generation. Unlike Kemp's approach, this method is not storing this metadata information. Because storing metadata information will lead to the incremental growth of the database, instead this research is describing the resources and a set of operators to access to the resources for automated navigation. The aim of this research was to keep the process dynamic and automated. This research has suggested proposition for local schema which will be used to store laboratory data and all related information associated with gene mutation data for variance analysis. The other

objective of this research is to initiate the multidatabase query based on image object keying, for example, to initiate the query by selecting a particular spot from gel electrophoresis image. Similar spot of protein images will be searched from gel electrophoresis databases for protein correlation. The result will be used to understand the molecular mechanism of diseases. This method of searching relevant data from heterogeneous database is a new dimension in database integration.

The thesis has presented a complete conceptual model for describing gene mutation. The reason for describing a complete data model for gene mutation data is to ensure that an indirect association of disease can be determined. Although other research works have looked at designing a conceptual schema for genome functional data, for example, Paton *et al.* (2000) and Okayama *et al.* (1998), but this research has been looking in designing models for gene mutation data for variance analysis. This particular research area has not been investigated by either Paton's group or by any other research groups who are working in this area.

Medigue *et al.*, (1999) and Achard *et al.* (2001) advocated for Object Database Management Systems instead of using conventional relational model because of available standardised query language and facilities to provide rich data model and data definition. They also highlighted the issues of inheritance, encapsulation, security granularity, concurrent usability and constraint declaration ability of OODBMS for data description. However, Kemper and Wiesner (2001) and Achard *et al.* (2001) showed in their research report that XML expresses more flexibility, simplicity and interoperability than OODBMS. However, the richness of OODBMS can be combined with the XML technology as a complementary to replace field/value based flat files used by XML Interchange Data Dumps (XIDD). XIDD ensures more flexibility for data transformation and creating wrapper. Achard *et al.* (2001) also emphasised that XIDD saves tedious and often sub-optimal parsing because it has its own standard and generic parsers.

The other aspect of this research is to design the conceptual model which will preserve the laboratory data. It will also determine how this local database would initiate the query linking the multidatabase for more meaningful information and analysis. The data model described in this thesis will extend the user participation in genetic variance analysis. The model will store all the laboratory results and other parameters required to analyse the genetic variance of disease phenotypes and

genotypes. The model will also be able to contribute its own data for knowledge sharing with other data resources on a common ground. Data collected from local component databases, as well as correlated data from different data sources, will provide an insight in the disease mechanism. Again, many researcher, (*e.g.*, Buneman *et al.*, 1995, Chen *et al.*, 1997 and Baker *et al* 1998) have investigated tools for distributed querying of biological sequence information sources. However, they have not addressed the issues of large scale cross-population data analysis where statistical data from population genetic analyses are referenced and the features of interest are visualised (Lancaste *et al.* 2003).

The research has looked into issues as how the multidatabase query can be initiated using image object keying for comparative study of the laboratory images and to visualise the information for complete understanding. It has included the images in query and in query results.

The thesis has presented the theoretical concept of image object keying for linking databases. An image retrieval method has been described in this thesis. The query is initiated by selecting gel electrophoresis spot from the image. Other conventional methods for matching, *e.g.*, template matching, cross correlation method *etc.*, have been found not to be effective and accurate for gel electrophoresis spot identification. This is because of its lack of deterministic identical spot. *pI* value (Isoelectric *pH*) and molecular weight of the protein are need to be determined for searching gel spot. Similar gel electrophoresis protein spots have been searched into the database on the basis of these values. However, this method will not work if either the molecular weight or any other parameter is unknown. This research has proposed an image based approach where the query is based on the image contents. The research has developed an image searching algorithm for molecular biology database query. This method has used both the image processing and geometric operation, which is quite unique in nature. The experimental results have shown the accuracy (>90%) and effectiveness in the gel electrophoresis image retrieval process which establishes the potential of this approach. The proposed method has assumed that the relevant pre-processing for noise and other enhancements if necessary have been already applied.

The proposed image based approach not only match the selected spot in the target image but it also retrieves the corresponding 3D images of protein. In the

context of gel image linking to resource databases, Lemkin (1997) developed a program (Flicker) for gel matching and 2DWG databases for gel description. Lemkin's (1997) work was based on hyper text linking of gel image spots to the resource databases. This involves linking multiple web pages. However, this research has used Resource Document Framework (RDF) for meta data description of the gels and then it has correlated these with the list of resources. This approach is unique and novel in the sense that multiple web pages does not need to be searched for 3D protein structure of selected gel image spot because it allows searching dynamically for the 3D structure of protein where the search is initiated by selecting the image object.

The linking of gel image protein spots to the resource databases is extended to correlate the gel image protein spot with multi-resources so that all the relevant data corresponding to the selected protein can be extracted. This research has used a case scenario where multidatabase query is used, utilising Protein Data Bank, Online Mendelian Inheritance in Man and Genome Data Bank. Many approaches which are available for multi-resources query (Kemp *et al.* 2000b, Von *et al.* 2000 and Kashyap and Sheth, 1998) do not provide any solution for using the web data which are the basis for loosely coupled databases. This research has highlighted the importance of web data for molecular biology research because of its rapid emergence and well acceptance. The integration of web data is far more challenging than integrating structured and standardised data since the web data does not follow any conventional data representation method.

Sahuguet and Azavant (2000) and Raghavan and Garcia-Molina (2001) attempted to integrate the web data. In their approach they developed wrappers using declarative specification language to describe the contents of the webs. This static approach can be replaced by the approach presented here which is not based on the content, it is based on the context. The reason of utilising context for integration is that it represents the roles of the participating objects instead the meaning of the objects. The model of subject domain is developed on the basis of purpose of integration, and not on the basis of the ontology. In this regard, Rector (2004) argued that classifying definitions and determining their consistency using multi-axial ontologies, which are often required in biomedical world for data retrieval, are extremely difficult to build. Thus, this research has attempted to develop an integration domain based on the contexts of the integration where object relationships

are prominent and it ignores the object value matching. A context graph is described in RDF to implement the approach.

The context graph description in RDF for resource mapping requires a theoretical explanation. Three essential parameters to describe the resource mapping are defined in this thesis. The triplet *has, provide* and *access* denote the relationships between the objects and the path to the resources for navigational purpose. The research has also suggested to build higher level of constructs for RDF to represent the complete contexts for web resource integration.

In this research, a framework for cooperative environment has been proposed instead of proposing any discrete and independent tools (Khan *et al.* 2002b). The local database is equipped with the following tools: metadata extractor, link mapper, query generator and result interpreter. These tools are transparent and hidden from the users. These tools maintain a cooperative environment to work on a common ground and to share common knowledge. The successful implementation of this proposed model depends on how these tools are used for effective search and retrieval operation.

The success of interoperability for any data model in biological domain depends on the ability of exchanging and interacting with other data models which are residing in external environment. To establish data exchange with external data resources, Steven and Miller (2000) described the role of CORBA for interoperability between bioinformatics resources. However, CORBA based applications have limitations for interoperable operations. CORBA employs protocols which add high overhead for transferring very small objects between resources and which is not efficient at all. One of the main drawbacks of CORBA is that its interface definition language (IDL) can not resolve semantic conflicts automatically. This research has thus employed XML and RDF as an alternative to CORBA and it has utilised the DOM interface for data exchange and data conversion as an alternative to CORBA IDL. DOM has the ability to deal with semistructured and very large data objects and it is effective to use for its openness.

The framework proposed here enables to compare protein data with other protein morphological data that are available in multiple molecular biology databases. Traditional methods do not give the options for this type of comparison and for linking 2D gel image data with the 3D protein structure and other text based protein data. The cooperative environment initiates the whole operation by a single response

which facilitates the local schema to look in other databases for comparing the protein morphology and other details for single disease. Experiments that were traditionally conducted on a single disease in a petri-dish can now be performed by using this integrated approach. This new approach for data integration complements the existing wealth of medical and molecular biology research.

8.3 Contribution of the Research

The main contribution of this research is its approach to deal with the iteroperability issues including the web data integration problem. The research has gained a wider insight in this research area which has lead to novel interpretations of issues relevant to molecular biology research. In this regard, the research has developed the concept of component databases for genetic disorder study in human. It has highlighted the concept of non-redundant schema integration (Khan *et al.* 2001a, 2001b) for integrating molecular biology databases. The research has proposed a method for linking databases based on image object keying which can be initiated by selecting a particular spot in a gel electrophoresis image. Finally, the research has suggested a framework for cooperative environment to integrate the web based molecular biology data. The research has emphasised on the automated identification of identical or similar proteins from public domain databases by selecting image object. It has developed a cooperative environment for database integration which has included a theoretical concept for describing a context and navigational approach for future database interoperability to share biological information. The strength of the cooperative environment for database integration lies in the fact that it is applied on loose binding of databases which has been ignored by the existing research in the domain of biological databases. Also the approach has shown how a discrete component database can be developed according to the need of a specific laboratory and which can be synchronised with the resource databases that can serve the biological community to a much greater extent. This establishes an alternative approach to the conventional consolidation of schemas into one standard global view. The research establishes the concept of independent and dedicated database within a specific framework which can aid the biological researchers to do a multidatabase query.

The work carried out in this research have been presented in computer based applications (for example, proceedings IEEE, 2001 2002 and 2003 and proceedings IIIS 2002) and in bioinformatics (for example, proceedings German conference of bioinformatics, 2001 and proceedings IEEE conference on Bioinformatics and Bioengineering, 2003) areas at international conferences.

Encouraging and positive comments regarding the approach taken to address the 'database integration problem' have been received from referees and from participants at these conferences. These comments include "the framework which have been described is interesting", "the problems which have been identified have not been addressed before", "replacing SQL scripts with image based searching is quite unique", "the approach need for protein spot identification is complementary to present approaches", *etc.*

8.4 Future Work

This Ph.D research has contributed to the understanding of heterogeneity of database schemas. It has described the diversity of molecular biology schemas and schema constructs. It has also looked at the concepts of database integration and framework for cooperative environment. The research aims to establish a functional prototype system which will use the image object keying concept to initiate the query and to integrate the heterogeneous biological resources for more meaningful information. The research has also aimed to look at the morphological differences of protein structures which will help to reveal the underlying mechanism of diseases. The research has established a cooperative environment where all the tools will work coherently and will generate code for multi-database query. A prototype model of this cooperative environment has been implemented in a small scale to synchronize with other biological resources. Research in this area is an on-going and an iterative process. So, the implementation of the framework and the prototype in real-time environment still needs to be implemented fully to see to what extent the model would work in such dynamic environment. The results obtained from this implementation would enable the research community to determine what other conditions, parameters or issues may need to be addressed within the proposed prototype system.

The present prototype model has not addressed the following issues in depth.

- Technical heterogeneity: it can not resolve automatically operating systems, access mechanism and hardware dependence of data.

- Structural heterogeneity: data needs to be converted in XML or in RDF representation from its original data model before taking part into multidatabase query.

These issues raise scopes for further research in the following areas.

Tool support

The design and implementation of the approach need to be supported by the tools as proposed in this research. These supporting tools should have the capability for locating sources automatically and extracting the metadata from the source model for accessing the database. The source model most often does not have the heterogeneity feature because of the hardware platform, operating systems or security aspects. Appropriate tools are also needed for generating the wrapper automatically in order to overcome the tedious definition of resources.

Diversified gel images

The gel images that are stored in public domain databases often have confusing or noisy spots on the gels because often they have not been pre-processed. Spots which are not clearly labelled can produce faulty correspondence to 3D structure of protein image. So, image spot diversification needs to be resolved prior to query initiation process.

Query capability

The current query initiation approach is based on image object selection. This query initiation method results in extracting the corresponding data elements. However, it does not provide the option for relational or object oriented database searching. This option can be included by constructing the appropriate context model and by developing an user friendly interface.

Context propagation

Further research can be initiated to overcome overlapping context description. Describing context for different meaning can cause context redundancy, so further research is needed to link one context with different meaning. A list of vocabulary can be developed in natural language to represent the context.

Result representation

Current presentation of the result is purely textual based and it does not accept any user interaction. Next part of the research can look into the data visualisation area

by implementing an interface which will enable the user to interact with visual data for automated comparison.

Prototype evaluation and system development

The prototype model has not been tested in a wider context, instead, a case has been chosen to verify the framework. So, this can be evaluated further in the next phase of the research. This would include collecting more results and feedback from practitioners and academics. This phase would also test the usefulness and the usability of the prototype by using expert and observational evaluations techniques. A user group will determine the usability and an expert group will evaluate the usefulness of the framework. The outcome from these phases will be used for further system development.

8.5 Concluding Remark

This chapter has summarised the Ph.D work and the usefulness of this research. The chapter has reflected on how this thesis has met its objectives and goals. It has also highlighted the findings of other research works, their limitations and the issues that have not been addressed yet. This research has focussed on those issues which have not been addressed yet. The chapter has also highlighted the research contribution in the field of bioinformatics and computer based applications. A list of accepted papers have been attached in Appendix B.

The thesis has contributed towards the solution of web based data integration. This is achieved by describing a context graph which represents semantics and purpose of integration. The research has also described a graph for automated navigation to reach to the data resources. This virtual integration of loosely coupled data, compare to physical data consolidation, *i.e.* data warehousing, data federation, is complex in nature and hence it requires more focus in this area. The framework, which as proposed here is an attempt to bridge the heterogeneity of web data. The encouraging results will provide a considerable input in system evolution which is independent of data construction and autonomous.

Finally, integrating data in independent and transparent manner is crucial as it is evident from many users community that integrating data by standardising or by data abstracting does not bring any benefits to the users community because of its high building and maintenance cost. So it is necessary that the proposed method as described here is built upon these concepts.

References

Abola, E.E., Bernstein F.C., and Koetzle T.F. (1998); Computational molecular biology. In Lesk A.M., (ed.), Sources and methods for sequence analysis; Oxford University Press, Oxford; pp. 69-81.

Achard, F., Vaysseix, G. and Barillot, E. (2001); XML, Bioinformatics and data integration; Bioinformatics, vol. 17, no. 2; pp. 115-125.

Alt, H. Behrends, B. and Blomer, J. (1995); Approximate matching of polygonal shapes; Ann. Math. Artificial Intelligence, vol. 13; pp. 251-266.

Alt, H. and Guibas., L. J., (2000); Discrete geometric shapes: matching, interpolation, and approximation; In Sack, J. R. and Urrutia, J. (ed), Handbook of computational Geometry, Elsevier Science Publishers, Amsterdam; pp121-153.

Alt, H., Brab, P., Godau, M., Knauer, C. and Wenk, C. (2001); Nearest neighbour search in Hausdorff distance pattern spaces; Technical Report, University of Berlin, Department of Mathematics and Informatics.

Appel, R.D., Vargas, J.R., Palagi, P.M., Walther, D. and Hochstrasser, D.F., (1997); Melanie II, a third-generation software package for analysis of two-dimensional electrophoresis images: II Algorithms; Electrophoresis, vol. 18; pp. 2735-2748.

Anderson, N.L., Taylor, J., Scandora, A.E., Coulter, B.P. and Anderson, N.G., (1981); The TYCHO system for computer analysis of two-dimensional gel electrophoresis patterns; Clinical Chemistry, vol. 27; pp.1807-1820.

Baker, P., Brass, A., Bechhofer, S., Goble, C., Paton, N. and Stevens, R. (1998); TAMBIS-transparent access to multiple biological information sources; In Proceedings of the International conference on Intelligent Systems for Molecular Biology, AAAI Press; pp. 25-34.

Ballard, D.H. (1981); Generalizing the Hough Transform to detect arbitrary shapes; In Pattern Recognition, vol. 13, no. 2; pp. 111-122.

Bairoch, A. and Boeckmann B. (1993); The SWISS-PROT protein sequence data bank, recent developments; Nucleic Acids Res. Vol. 21; pp. 3093-3096.

Barnsley, M. (ed) (1988); Fractals everywhere; 2nd Edition, Academic Press, Boston.

Baxevanis, D. (2001); The molecular biology database collection: an updated compilation of biological database resources; Nucleic Acids Research. vol. 29, no.1; pp. 1-10.

Bayat, A., (2002); Science medicine and the future; British Medical Journal (BMJ), vol. 324; pp. 1018-1022.

Benson, D., Lipman, D.J. and Ostell, J. (1993). NICBI/GenBank; Nucleic Acids Research, vol. 21; pp. 2963-2965.

Berendt, B. and Spiliopoulou, M. (2000); Analysis of navigation behaviour in websites integrating multiple information systems; The VLDB Journal Springer-Verlage Press, vol. 9; pp. 56-75.

Bernstein, F.C., Koetzle, T.F., Williams, G.J.B., Mayer, E.F., Bryce, M.D., Rodgers, J.R., Kennard, O., Shimanouchi, T. and Tasumi, M. (1977); The Protein Data Bank: A computer based archival file for macromolecular structures; Molecular Biology, vol. 112; pp. 535-542.

Bettens, E., Scheunders, P., Van Dyck, D., Moens, L., Van Osta, P. (1997); Computer analysis of two-dimensional electrophoresis gels: a new segmentation and modelling algorithm; Electrophoresis, vol. 18; pp. 792-798.

Blake, J. (1995); Inter-Connection of biological databases: exploring different Levels of molecular biology database federation; In Second Meeting on interconnection of Molecular Biology Databases, Cambridge, United Kingdom. http://www-genome.wi.mit.edu/informatics/abstracts.html.

Booch, G., Rumbaugh, J. and Jacobson, I. (eds) (1999); The Unified Modelling Language user guide. Addison-Wesley.

Brab, P. and Knauer, C. (2001); Nearest neighbour search in Hausdorff distance; Technical Report, University of Berlin, Department of Mathematics and Informatics.

Bright, M.W., Hurson, A.R., and Pakzad, H.A. (1992); Taxonomy and current issues in multidatabase systems; IEEE Computer, vol. 25, no. 3; pp 50-59.

Buneman, P., Davidson, S., Hart, K., Overton, C. and Wong, L. (1995); A data transformation system for biological data sources; In Proc. 21st Very Large Database, Kaufmann, M. (ed); pp. 158-169.

Busse, S., Kutsche, R., Leser, U. and Webber, H. (1999); Federated information systems: concepts, terminology and architectures; Technische Universitat Berlin, Forschungsberichte des Fachbereichs Informatik 99.

Busse, S., Kutsche, R. and Leser, U. (2000); Strategies for the conceptual design of federated information systems; In Proc. Third international workshop on Engineering of Federated Information system; pp. 23-32.

Canny, J. (1986); A computational approach to edge detection; IEEE Trans. Pattern Analysis and Machine Intelligence. PAMI-8; pp. 679-698.

CAROL (1998); URL: http://gelmatching. Inf.fu-berlin.de; CAROL software system for matching 2D GEL Images.

Chen, I. A., Kosky, A., Markowitz, V.M., Szeto, E. (1995); OPM*QS: The Object-Protocol Model Multidatabase Query System. Documentation: Technical Report LBNL-38181.

Chen, I. A. and Markowitz, V.M.(ed) (1995); An overview of the Object-Protocol Model (OPM) and OPM Data Management Tools, Information Systems, Pergamon Press.

Chen, I. A., Kosky, A., Markowitz, V.M. and Szeto, E. (1997); Constructing and maintaining scientific database views in the framework of the Object Protocol Model; In proc. IEEE Scientific and Statistical Datbase Management, IEEE press. Pp. 237-248.

Cheung, K., Kumar, A., Snyder, M. and Miller, P. (2000); An integrated web interface for large-scale characterisation of sequence data; Functional Integrated Genomics, Springer-Verlag Publication; Pp. 70-75.

Cheung, K., Liu, Y., Kumar, A., Snyder, M., Gerstein, M. and Miller, P. (2001); An XML application for genomic data interoperation; 2nd IEEE Bioinformatics and Bioengineering Symposium. IEEE Press; pp. 97-103.

Chui, H. and Rangaranjan, A., (2000); A New Algorithm for non-rigid point matching; In IEEE Conference on Computer Vision and Pattern Recognition (CVPR) vol. 2; pp. 44-51.

Cluet, S., Veltri, P. and Vodislan, D. (2001); Views in a large XML repository; In Proc. 27th Very Large Database Conference; Pp 271-280.

Connolly and Begg (ed.) (2002); Database Systems; Addison Wesley Publisher. ISBN: 0-210-70857-4.

Cooper, D. N. and Krawczak, M. (ed) (1993); Human gene mutation, BIOS Scientific Publisher, Oxford.

Cooper, D. N., Ball V. E. and Krawczak, M. (1999); The human gene mutation Database; Nucleic Acid Research; vol. 26, no. 1-3; pp. 285-287.

Cornell, M., Paton, N.W., Wu, S., Goble, C.A., Miller, C.J., Kirby, P., Eilbeck, K., Brass, A., Hayes, A., and Oliver, S.G. (2001); GIMS: A data warehouse for storage and analysis of genome sequence and functional data; Proc. 2nd IEEE International Symposium on Bioinformatics and Bioengineering (BIBE), IEEE press; pp. 15-22.

Cozza, S., Reed, E. C., Salit, J., Chang, W., Marr, T. (1994); Genome Topographer: A next generation genome database system (Abstract), presented at the meeting on Genome Mapping and Sequencing; Cold Spring Harbor Laboratory, Cold Spring Harbor. URL: http://www.ornl.gov/sci/techresources/Human_Genome

Cuticchia, A.J., Fasman, K.H., Kingsbury, D.T. Robbins, R.J. and Pearson, P.L., (1993); Human genome database Anno 1993; Nucleic Acids Research, vol. 21, no. 13; pp. 3003-3006.

Davidson, S. B., Overton, C. and Buneman, P.J. (1995); Challenges in integrating biological data source; J. Computational Biology; vol: 2, no: 4; pp. 557-572.

Davidson, D., Bard, J., Brune, R., Burger, A., Dubreuil, C., Hill, W., Kaufman, M, Quinn, J., Stark, M. and Baldock, R., (1997); The mouse atlas and graphical gene-expression database; Seminars in Cell and Developmental biology; vol. 8; pp. 509-517.

Davidson, S., Buneman P., Crabtree, J., Tannen, V., Overton, C., Wong, L. (1999); Integrating biomedical data and analysis packages; In: Letovsky S, (ed). Bioinformatics databases and systems. Boston, Kluwer Academic Publisher; pp. 201-212.

Davies, A.J., Brandli, W.A., Hunter, D. and Nienminen, P., (1997); Design Considerations for Small, Special-System Developmental Databases; Seminars in Cell and Developmental Biology; vol 8; pp. 519-525.

DOM (2000); URL: http://www.w3.org/TR/2000/WD-DOM-Level-1-20000929/ DOM Specification Level 1.

Donnelly, B., (2003); Data integration technologies: An unfulfilled revolution in the drug discovery process; Biosilico; vol. 1, no. May; pp 59-63.

Duda, R.O. and Hart, P.E., (1972); Use of the Hough Transofrmation to detect lines and curves in pictures; Communication of the ACM; vol. 15, no. 1; pp. 11-15.

Edgar, G.A., (ed) (1995); Measure, topology, and fractal geometry; Springer UTM.

Ehrenmann, M., Ambela, D., Steinhaus, P. and Dillmann, R., (2000); A comparison of four fast vision based object recognition methods for programming by demonstration applications; International Conference on Robotics and Automation (ICRA), April 24-28, USA; pp.1862-1867.

Eidhammer, I., Jonassen, I. and Tylor, W. R., (2000); Structure comparison and structure patterns; Journal of Computational Biology, Mary Ann Liebert, Inc Publisher; vol.7, no. 5; pp 685-716.

Efrat, A., Hoffmann, F., Kriegel, K., Schultz, C. and Wenk, C. (2001); Geometric

algorithm for the analysis of 2D electrophoresis Gels; Proc. 5th Int. Conference on Computational Molecular Biology, RECOMB, Celera Genomics; pp.114-123.

El-Beltagy, S.R., Hall, W, Roure, D.D. and Carr, L. (2001); Linking in context; Conference on hypertext and hypermedia; Proc. 12th ACM Conference on hypertext and hypermedia, Denmark; pp. 151-161.

EMBL (2003); URL: http://www.ebi.ac.uk, EMBL, Release 77

Etzold, T., and Argos, P. (1993); Transforming a set of biological flat file libraries to a fast access network; Computer Applications of Biosciences, vol. 9, no. 1; pp 58-64.

Etzold, T., Ulyanov, A. and Argos, P. (1996); SRS: information retrieval system for molecular biology data banks: Methods in Enzymology, Academic Press; vol. 266, New York; pp. 114-128.

ExPASY (2000); Expert Protein Analysis System; URL: http:\\www.expasy.org

Fasman, K.H., Letovsky, S.I., Cottingham, R.W., and Kingsbury, D.T. (1996); Imrpovements to the GDB Human Genome Database; Nucleic Acids Research, vol. 24, no. 1; pp. 57-63.

Freidman, M., Levy, A. and Millstein, T. (1999); Navigational plans for data integration; In proc. Sixteenth National Conference on Artificial Intelligence and Eleventh Conference on Innovative Application of Artificial Intelligence, AAAI-Press, ISBN: 0-262-51106-1; pp. 67-73.

Garrels, J. I., Farrar, J.T. Burwell, C. B., in: Celis, J. E. and Bravo, R.(ed.), (1984); Two- dimensional gel electrophoresis of proteins; Academic Press; pp.37-91.

Geihs, K (2001); Middleware challenge ahead; In 5th International Workshop The Internet Challenge: Technology and Applications; Berlin, Germany, Kluwer Publishers; pp. 24-31.

Gieger, C., Deneke H., and Fluck, J., (2003); The future of text mining in genome-based clinical research; Biosilico, vol 1, no. 3; Pp. 97-102.

Gonzalez, R. and Woods, R. (ed) (1992); Digital image processing; Addison Wesley; pp. 414 - 428.

Goto, S., Akiyama, Y., and Kanehisa, M. (1995); Link DB: A database of cross links between molecular biology databases; In Second Meeting on Interconnection of Molecular Biology Databases, Cambridge, United Kingdom, 1995. URL: http://www.genome.ad.jp/kegg/docs/mimbd95b.pdf

Halevy, A.Y., Ives, Z,G., Mork, P. and Tatarinov, I. (2003); Piazza: data management infrastructure for semantic web applications; 12th International WWW conference. ACM Publishers.

Hough, P.V.C., (1962); Method and means for recognising complex patterns; US patent 3,069,654.

Huck, G., P. Frankshuser, K. Aberer and Neuhold, E. (1998); JEDI: extracting and synthesising information from the Web; In proc. 6th International Conference on Cooperative Information Systems; pp. 32-43.

Huttenlocher, D.P., Klanderman, G. and Rucklidge, W. (1993); Comparing images using the Hausdorff distance; IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 15; Pp. 850-863.

Illingworth, J and Kittler, J. (1988); A survey of the Hough Transform; Computer vision, graphics and Image Processing, vol. 44, no. 1; pp. 87-116.

Jain, E., (2002); Distributed computing in Bioinformatics; Applied Bioinformatics, vol. 1, no. 1; pp. 13-20.

Jugfer, K. and Rodriguez-Tomes, P. (1998); Mapplet: a CORBA based genome map viewer. Bioinformatics, vol. 14; pp. 734-738.

Karp, P.D., (1994); Report of the first meeting on interconnection of molecular biology databases (MIMBD 94); Stanford, California, August 1994. URL: http://www.sri.ai.com/people/pkarp/mimbd/mimbd-94.html.

Karp, P.D., (1995); Strategy for database interoperation; J. Comptutational Biology, vol. 2, no. 4; pp. 573-586.

Karp, P.D, (1996); Database links are a foundation for interoperability; Trends in Biotechnology, vol. 14; Pp 273-279.

Karp, P.D and Suzane, P., (1996); Integrated access to metabolic and Genomic Data; J. computational Biology, vol. 3, no. 1; pp. 191-212.

Karp, P.D. Riley, M., Paley, S.M., Pellegrini-Toole, A. and Krummenacker, M. (1998); EcoCyc: encyclopedia of *Escherichia coli* gene and metabolism. Nucleic Acids Research, vol. 26; pp. 50-53.

Kashyap, V. and Sheth, A. (1996); Semantic and schematic similarities between database object: a context –based approach; The Very Large Database journal Springer-Verlag, vol. 5, no. 4; pp. 276-304.

Kashyap, V. and Sheth, A. (1998); Semantic heterogeneity in global information systems: the role of Metadata, Context and Ontologies; In cooperative information systems. Papazoglou, M.P. and Schlageter, G (ed), Academic Press, San Diego. pp. 139-178.

Kemp, G.J.L., Dupont, J. and Gray, P.M.D. (1996); Using the functional data model to integrate distributed biological data sources; In Svensson, P. and French, J.C. (eds), Proc. Eighth International Conference on Scientific and Statistical Database Management, IEEE Computer Society Press; pp. 176-185.

Kemp, G.J.L, Agelopoulos, N. and Gray, P. (2000a); A schema-based approach to building a Bioinformatics database federation; Proc. IEEE International symposium on Bioinformatics and Biomedical Engineering. IEEE Computer science press; pp. 13-20.

Kemp, G.J.L., Robertson, C.J., Gray, P.M.D. and Angelopoulos, N. (2000b); CORBA and XML: design choices for database federations; In Lings, B. Jeffery, K (eds.), Advances in Databases, Proc. of the seventeenth British National Conference on Databases (BNCOD17), Springer Verlag, Barlin; pp. 191-208.

Kemper, A. and Wiesner, C. (2001); Hyper Queries: dynamic distributed query processing on the internet; Proc. Very Large Database Conference; pp. 551-560.

Khan, N. and Rahman, S. (2001); A conceptual object model of gene mutation data; Proc. German Conference on Bioinformatics. Germany; ISBN:3-00-008114-3; pp. 187-190.

Khan, N., Rahman, S. and Clarkson, G. T. (2001); An approach to develop human gene disorder database for intelligent variance analysis of genes and its products; 12th International conference on Database and Expert System (DEXA, 2001). Germany, Munich. Proc. IEEE Computer society press; pp. 301-305.

Khan, N and Rahman, S. (2002a); Object modelling of gene mutation data for variance analysis; 6th World Conference on Systemics, Cybernatics and Informatics (SCI in Medical and Biology); Proc. International Institute of Information Systems (IIIS), USA; pp.301-306.

Khan, N and Rahman, S. (2002b); A cooperative environment for genetic variance analysis using component database for database integration; 15th IEEE symposium of computer based medical application. Proc. IEEE computer society press; pp. 365-368.

Khan, N and Rahman, S. (2003); A new approach to detect Similar proteins for 2D gel mages; 3rd International IEEE Conference on Bioinformatics and Bioengineering; pp. 182-189.

Khan, N, Rahman, S. and Stockman, T. (2003); Integration of biological resources using image object keying; 16th IEEE Computer Based Medical System. IEEE Press; pp. 16-21.

Kingsbury, D. T. (1993); Report of the Invitational DOE workshop on Genome Informatics, 26-27 April. URL: http://www.ornl.gov/sci/techresources/Human_Genome/publicat/miscpubs/bioinfo/contents.shtml

Krawczak, M., Ball, V. E., Fenton, I., Stenson, D. P., Thomas, N. and Cooper, N. D. (2000); Human gene mutation database – A biomedical information and research resource; Human mutation, vol. 15; pp. 45-51.

Kriegel, K., Seefeldt, I., Hoffmann, F., Schultz, C., Wenk, C., Regitz-Zagrosek, V. and Fleck, E. (2000); An alternative approach to deal with geometric uncertainties in computer analysis of two-dimensional electrophoresis gels; Electrophoresis, vol. 21; pp.2637-2640.

Kutsche, R. and Sunbul, A. (1999); A meta-data based development strategy for heterogeneous, distributed information systems; In Proc. 3rd IEEE Metadata Conference; URL: http://www.computer.org/proceedings/meta/1999/papers/60/rkutsche.html

Lancaster, A., Nelson M. P., Meyer, D. and Thompson, G. (2003); PyPop: A software framework for population genomics: analysing large-scale multi-locus genotype data; Pacific Symposium on Biocomputing (PSB), vol. 8; pp. 514-525.

Leser, U. (1998); Maintenance and mediation in federated database; In proc. 8th Workshop on information technology and systems; pp. 187-196.

Lemkin, P.F. (1997); The 2DWG meta-database of 2-D electrophoretic gel images on the internet; Electrophoresis, vol. 18; pp. 2759-2773.

Lemkin, P. and Lipkin, L. E. (1981a); GELLAB: A computer system for 2D gel electrophoresis analysis. I. Segmentation and preliminaries; Computers and Biomedical Research, vol. 14; pp. 272-297.

Lemkin, P. and Lipkin, L. E. (1981b); GELLAB: A computer system for 2D gel electrophoresis analysis. II. Spot pairing; Computers and Biomedical Research, vol. 14; pp.355-380.

Lemkin, P.F., Lipkin, L.E. and Lester, E.P., (1982); Some extensions to the GELLAB 2D electrophoresis gel analysis system; Clinical Chemistry, vol. 28; pp.840-849.

Lester, E.P., Lemkin, P., Lipkin, L.E. (1981) New dimensions in protein analysis – 2D gels coming of age through image processing; Analytical Chemistry, vol. 53; pp.390-397.

Letovsky, S. (1995); Beyond the information maze; Journal of Computational Biology, vol. 2, no. 4; pp. 539-546.

Liu, B., Xia, Y. and Yu, P.S. (2000); Clustering through decision tree construction; in proc. ACM-CIKM; pp 20-29.

Lopez, A.M. (1999); Evaluation of methods for ridge and valley detection; IEEE Trans. Pattern Analysis and Machine Intelligence, PAMI-21; pp. 327-335.

Luscombe, N.M., Greenbaum, D., and Gerstein, M., (2001); What is Bioinformatics? A proposed definition and overview of the field; Method inform Med. vol. 40; pp. 346-58.

Manolescu, I., Florescu, D. and Kossmann, D. (2001); Answering XML queries over heterogeneous data sources; In. Proc. 27th Very Large Database, Italy; pp. 241-250.

Markowitz, V.M. (1995a); Heterogeneous molecular biology database, Journal of Computational Biology, vol. 2, no. 4; pp. 537-538.

Markowitz, V. M. (1995b); Coping with data modelling diversity; URL: http://gizmo.lbl.gov/DM_TOOLS/PAPERS/DMD/DMD.html

Markowitz, V.M and Ritter, O. (1995); Characterising heterogeneous molecular biology database systems; J. Computational Biology, vol. 2, no. 4; pp. 547-556.

Markowitz, V. M., Chen, I.A. and Kosky A. (1996); Theoretical and computational genome research; S. Suhai (ed), Plenum Publication.

Markowitz, V. M. and Topaloglou, T. (2001); Applying data warehouse concepts to gene expression data management; In Proc. Second IEEE symposium on Bioinformatics and Bioengineering, IEEE Press; Pp. 65-71.

Marr, D. and Hildreth, E. (1980); Theory of edge detection; Proc. Royal Society of London B 270; pp. 187-217.

Martin, C.R.A (2001); Can we integrate bioinformatics data on the web?; Trends in Biotechnology, vol. 19, no. 9; pp. 327-328.

McHugh, J., Abiteboul S., Goldman R., Quass D., and Widom J. (1997); Lore: A database management system for semi-structured data; In SIGMOD record, vol. 26, no. 3; pp. 54-66.

McKusick V.A. (eds) (1991); Catalogs of autosomal dominant, autosomal recessive, and X-linked phenotypes; Tenth Edition, Johns Hopkins University Press, Baltimore.

Medigue, C., Rechenmann, F., Danchin, A. and Viari, A. (1999); Imagene: an integrated computer Environment for sequence annotation and analysis; Bioinformatics, vol. 15; pp. 2-15.

Melanie (2000); URL: http://us.expasy.org/melanie/ Melanie Gel Matching Tools.

Mihaila, G.A., Raschid, L. and Tomasic, A. (2002); Locating and accessing data repositories with web Semantics; The Very Large Database Journal, vol. 11; pp. 47-57.

Miller, R.J., Ioannidis, Y.E., and Ramakrishnan, R., (1993); The Use of information capacity in schema integration and translation; Proc. 19th International Conference on Very Large Databases; pp. 120-133.

Mork, P., Halevy, A., Tarczy-Hornoch, P. (2001); A model for data integration systems of biomedical data applied to online genetic databases; In proc. American Medial Informatics Association Annual Symposium,Washington D.C. pp. 473-477.

Navathe, S. B., Elmasri, R. and Larson, J.A. (1986); Integrating user views in database design; IEEE Computer, vol. 9, no. 1; pp. 50-62.

Okayama, T., Tamura, T., Gojobori, T., Tateno, Y., Ikeo, K., Muyazaki, S., Fukami Kobayashi, K. and Sugawara, H., (1998); Formal design and implemntation of

an improved DDBJ DNA database with a new schema and object oriented library; Bioinformatics, vol. 14, no. 6; pp. 472-478.

Ouksel, A. and Naiman, C. (1993); Coordinating context building in heterogeneous information systems; J Intelligent Information System, vol. 3; pp. 151-183.

Panek. J. and Vohradsky, J., (1999); Point pattern matching in the analysis of two dimensional gel electropherograms; Electrophoresis, vol. 20; pp. 3483-3491.

Paton, N.W., Stevens, R., Baker, P., Goble, C.A., Bechhofer, S. and Brass, A. (1999); Query processing in the TAMBIS Bioinformatics source integration system; In 11th International Conference on Scientific and Statistical Database Management, Proc. IEEE Computer Society Press; Pp. 138-147.

Paton, W.N., Khan, S.A., Hayes, A., Moussouni F., Brass, A., Eilbeck, K., Goble, A.C., Hubbard, S.J. and Oliver, S.G., (2000); Conceptual modelling of genomic information; Bioinformatics, vol. 16, no. 6; pp. 548-557.

Pleibner, K., Hoffmann, F., Kriegel, K., Wenk, C., Wegner, S., Sahistrom, A., Oswald, H., Alt, H. and Fleck, E. (1999); New algorithmic approaches to protein spot detection and pattern matching in two-dimensional electrophoresis gel databases; Electrophoresis, vol. 20; pp. 755-765.

Pratt, W.K (ed.), (2001); Digital image processing; A Wiley Interscience Publication. ISBN: 0-471-37407-5.

Prehm, J., Jungblut, P., Klose, J. (1987); Analysis of two dimensional protein patterns using a video camera and a computer; Electrophoresis, vol. 8; pp. 562-572.

Prewitt, J.M.S. (1970); Object enhancement and extraction in picture processing psychopictorics; Lipkin, B.S. and Rosenfeld, A., (eds). Academic Press, New York.

QBIC (2002); Query By Image Content by IBM. URL: http://wwwqbic.almaden.ibm.com/.

Ram, S., Park, J. and Hwang, Y. (2002); CREAM: A mediator based environment for modelling and accessing distributed information on the web; In proc. British National Conference on Database (BNCOD),. Springer, vol. 2405, pp 58-61.

Rao, D.C., Keats, B.J.B., Morton, N.E., Yee, S. and Lew, R. (1978); Viability of human linkage data; Ann. J. Human Genetics, vol. 30. Pp. 516-529.

Raghavan, S. and Garcia-Molina, H. (2001); Crawling the hidden web; In proceedings 27th Very Large Database conference, Italy; pp.129-138.

RDF (1999); URL: http://www.w3.org Resource Description Framework (RDF) Model and Syntax Specification.

Rector, A. (2004); Defaults, context and knowledge: Alternatives for OWL-indexed knowledge bases; Pacific Symposium on Biocomputing (PSB); http://www-smi.stanford.edu/projects/helix/psb04/rector.doc

Ritter, O. (1994); The Integrated Genomic Database; In Computational Methods in Genome Research; Suhai, S., (ed), Plenum, pp. 57-73.

Robbins, R.J. (1994); Report of the invitational DOE workshop on genome informatics; Baltimore, Maryland. Genome Informatics I: Community database. J. Computational Biology, vol. 1; pp. 173-190.

Robbins, R.J., (1996); Bioinformatics: essential infrastructure for global biology; J. Computational Biology, vol. 3, no. 3; pp. 465-478.

Roberts, L.G., (1965); Machine perception of three-dimensional solids; in Optical and Electro-Optical Information, Proc. Tippett, J. T., *et al.* (eds). MIT press, Cambridge, MA, pp. 159-197.

Rodrigo, A. (2002); Editorial Foreword; Applied Bioinformatics, vol.1, no.1, pp. 1-2.

Roure, D. D, Hall, W., Reich, S., Hill, G., Pikrakis, A. and Stairmand, M. (2001); MEMOIR-an open framework for enhanced navigation of distributed information; Journal of Information Processing and Management, PERGAMON press, vol. 37; Pp.53-74.

Sahuguet, A. and Azavant, F. (1999); Building light-weight wrappers for legacy web data-sources using W4F; 25th conference on very large database systems Edinburgh, UK. pp. 738-741.

Sahuguet, A. and Azavant, F. (2001); Building intelligent web applications using lightweight wrappers; Data and Knowledge Engineering, Elsevier Science Press, vol. 36, no. 3; Pp.283-316.

Sarker, I. N., Cantor, M.N., Gelman, R., Hartel, F., Lussier, Y.A. (2003); Linking biomedical language information and knowledge resources in the 21st Century; In Pacific Symposium in Biocomputing (PSB); vol. 8; Pp. 415-426.

Schonbach, C., Kowalski-Saunders, P. and Brusic, V., (2000); Data warehousing in molecular biology; Briefings in Bioinformatics, vol. 1, no. 2, ISSN: 1467-5463; Pp. 190-198.

Sean, K (ed) (1994); Data warehousing: The route to mass customisation; John Wiley and Sons. ISBN: 0-471-95082-3.

Shaker, R, Mork, P., Barclay, M. and Tarczy-Hornoch, P. (2002); A rule driven bi-directional translation system for re-mapping queries and result sets between a mediated schema and heterogeneous data sources; In Poc. American Medical Informatics Association, Annual Symposium; pp. 692-696.

Shanmugasundaram, J., Kiernan, J., Shekita, E., Fan, C., Funderburk, J. (2001); Querying XML views of relational data; In 27th Very Large Database conference; pp. 261-270.

Sheth, A, Kashyap V. (1992); So far schematically yet so near semantically; In Proc. IFIP TC2/WG2.6 conference on semantics of interoperable database systems. IFIP Transactions A-25, Holland. Pp. 283-312.

Sheth, A.P., and Larson, J.A. (1990); Federated database systems for managing distributed, heterogeneous, and autonomous databases; ACM Computing Survey, vol. 22, no. 3; pp. 183-236.

Sheu, P, C-Y, Cummings, B., Cotman, C, Chubb, C, Hu, L., Wang, T., Johnson, J., Mobley, S., Stitch, T. and Inagaki, Y. (2000); An object relational approach to biomedical database; In 1st IEEE Conference on Bioinformatics and Bioengineering; Pp. 91-71.

Siepel, A., Farmer, A., Tolopko, A., Zhuang, M., Mendes, P., Beavis, W., and Sobral, B. (2001); ISYS: a decentralized, component-based approach to the integration of heterogeneous bioinformatics resources; Bioinformatics, vol. 17, no. 1; pp. 83-94.

Smilansky, Z. (2001); Automatic registration for images of two-dimensional protein; Electrophoresis, vol. 22; pp. 1616-1626.

Sondag, P. (2001); The semantic web paving the way to the knowledge society; Proc. 27th International Conference on Very Large Databases, ISBN 1-55860-804-4; pp. 16.

Steven, R. and Miller, C. (2000); Wrapping and interoperating bioinformatics using CORBA; Briefings in Bioinformatics, vol. 1, no. 1; pp 9-21.

Stevens, R., Goble C., Baker, P. and Brass, A. (2001); A classification of tasks in Bioinformatics; Bioinformatics, vol. 17, no. 2; Pp 180-188.

SWISS-2D (2000);URL: http:\\www.expasy.org/ch2d/, Swiss-2DPAGE, Nucleic Acid Research, vol. 28; pp.286-288.

Tekalp, A. M. (2000); Video segmentation; In Al Bovik (eds.), Handbook of Image Video Processing; Academic Press Series in Communication, Networking and Multimedia; Pp 383-399.

Vogel, F. (1990); Mutation in man. In: Principles and Practices of medical genetics, Emery, A.E.H. and Rimoin, D. (eds). Churchill Livingstone, Edinburgh, vol 1.

Von, V. (2000); Query Planning in Mediator Based Information System, Ph.D thesis. Technischen Universitat Berlin.

Wadler, P. (2000); A formal semantics of patterns in XSLT; Markup Technologies; URL: citeseer.nj.nec.com/wadler00formal.html

Weber, G., Knipping, L., Alt, H., J. (1994); An application of point pattern matching in astronautics; Symbolic Computation, vol. 17; pp 321-340.

Wheeler, D.L., Church, D.M., Lash, A.E., Leipe, D.D., Madden, T.L., Pontius, J.U., Schuler, G.D., Schrimi, L.M., Tatusova, T.A., Wagner, L. and Rapp., B.A. (2001); Database resources of the National Centre for Biotechnology; Nucleic Acid Research, vol. 29, no. 1; Pp. 11-16

Wiederhold, G. (1992); Mediators in the architecture of future information systems; IEEE Computer, 25(3); pp38-49.

Wong, R.K. and Shui, W. M. (2001); Utilizing multiple Bioinformatics information sources: An XML database approach; In Proc. 2ND IEEE Symposium on Bioinformatics and Bioengineering. IEEE Press; Pp 73-79.

Xie, G., Demarco, R., Blevins, R. and Wang, Y. (2000); Storing biological sequence databases in relational form; Bioinformatics, vol. 16, no. 3; pp 288-289.

XML (2000); Extensible Markup Language (XML) 1.0, Second Edition, T. Bray, J. Paoli, C.M. Sperberg-McQueen and E. Maler, (ed) World Wide Web Consortium. URL: http://www.w3.org/TR/REC-xml.

XSLT (1999); XSL Transformation, Version 1.0, W3 Recommendations 1999; James Clark (ed.) URL: http://www.w3.org/TR/xslt.

Appendix A

Glossary

Abstract Syntax Notation One (ASN.1) is a formal language for describing messages which is to be exchanged among an extensive range of applications involving the Internet, intelligent network, cellular phones, *etc.* in an abstract form.

AceDB abbreviation for A *Caenorhabditis elegans* Database. Containing data from Caenorhabditis Gentics Centre, C. elegans genome project and the worm community. AceDB is also the name of the generic genome database software in use by an increasing number of genome projects. AceDB databases are available for the following species: *C. elegans*, Human Chromosome 21, Human Chromosome X etc.

Agarose Gel is polyacrylamide gels that used as media for gel electrophoresis because they are chemically inert and are readily formed by the polymerization of acrylamide.

Allele is one of several alternative forms of a gene occupying a given locus on a chromosome.

Amber codon is the nucleotide triplet UAG, one of three codons that cause termination of protein synthesis.

Amber mutation describes any change in DNA that creates an amber codon at a site previously occupied by a codon representing an amino acid in a protein.

Amplification refers to the production of additional copies of a chromosomal sequence, found as intra chromosomal (within chromosome) or extra chromosomal (outside of the chromosome) DNA.

Annealing is the pairing of complementary single strands of DNA to form a double helix.

ANSI/SPARC's layered model of database architecture comprises a physical schema, a conceptual schema and user views.

Autoradiography detects radioactively labelled molecules by their effect in creating an image on photographic film.

Base pair (bp) is a partnership of A with T or of C with G in a DNA double helix; other pairs can be formed in RNA (ribo-nucleic acid) under certain circumstances. Distance along DNA is measured in bp.

BLAST abbreviation for Basic Local Alignment Search Tool. This is the most popular program for searching sequence in databases. It is very fast and able to search DNA and protein sequences from databases.

cDNA is a single stranded DNA complementary to an RNA, synthesized from it by reverse transcription *in vitro*.

Chromosome is a discrete unit of the genome carrying many genes. Each chromosome consists of a very long molecule of duplex DNA and an approximately equal mass of proteins. It is visible as a morphological entity only during cell division.

Clone describes a large number of cells or molecules identical with a single ancestral cell of molecule.

Codon is a triplet of nucleotides that represents an amino acid or termination signal.

Crystallographic imaging techniques employed x-ray diffraction, x-ray scattering, x-ray absorption/emission spectroscopy, and advanced biomedical imaging techniques such as diffraction enhanced imaging (DEI) to determine molecular structure.

Cytogenetics is the study of the physical appearance of chromosomes.

Deductive capability is the technique which highlights the fact that the system is able to make deductions from facts stored in the database using rules stored in the database.

Deletions are generated by removal of sequence of DNA, the regions on either side being joined together.

Denaturation of protein describes its conversion from the physiological conformation to some other (inactive) conformation.

Direct repeats are identical sequences present in two or more copies in the same orientation and in the same molecule of DNA; they are not necessarily adjacent.

Erythroleukemia is a particular type of tumour in human.

Exon is any segment of an interrupted gene that is represented in the mature RNA product.

Filter hybridization is performed by incubating a denatured DNA preparation immobilized on a nutrocellulose filter with a solution of radioactively labelled RNA or DNA.

Flanking region is the DNA sequences extending on either side of a specific locus or gene.

Frameshift mutations arise by deletions of insertions that are not a multiple of 3bp (base pair); they change the frame in which triplets are translated into protein.

Frictional coefficient depends on both the mass and shape of the migrating molecule and the viscosity of the medium.

Gene is the segment of DNA involved in producing a polypeptide chain; it includes regions preceding and following the coding region as well as intervening sequences between individual coding segments.

Genetic code is the correspondence between triplets in DNA and amino acids in protein.

Genotype is the genetic constitution of an organism.

Hybridization is the pairing of complementary RNA and DNA strands to give an RNA-DNA hybrid.

Initiation codon is a triplet sequence that initiate protein synthesis.

Insertions are identified by the presence of an additional stretch of base pairs in DNA.

Intron is a segment of DNA that is transcribed, but removed from within the transcript by splicing together the sequences (exon) on either side of it.

Inversion is a chromosomal change in which a segment has been rotated by 180^0 relative to the regions on either side and reinserted.

MEDLINE is bibliographic database covering the fields of medicine, nursing, dentistry, veterinary medicine, and the preclinical sciences. Journal articles are indexed for MEDLINE, and their citations are searchable.

Modification of DNA or RNA includes all changes made to the nucleotides (A nucleotide consists of a nitrogenous base, a sugar, and one or more phosphate groups) after their initial incorporation into the polynucleotide chain (a number of nucleotides in a chain).

mRNA is abbreviation of messager RNA

Mutation describes any change in the sequence of genomic DNA.

Mutation frequency is the frequency at which a particular mutant is found in the population.

Mutation rate is the rate at which a particular mutation occurs, usually given as the number of events per gene per generation.

Network mediated program deals with communication issues, management, and implementation problems raised by network-based communication services (e.g., World Wide Web, e-mail, conferencing systems, etc.)

Nonsense mutation is any change in DNA that causes a codon to replace a codon representing an amino acid.

Northern blotting is a technique for transferring RNA from an agarose gel to a nitrocellulose filter (a paper which can bind DNA) on which it can be hybridized to a complementary DNA.

PCR (polymerase chain reaction) describes a technique in which cycles of denaturation, annealing with primer, and extension with DNA polymerase are used to amplify the number of copies of a target DNA sequence.

Phenotype is the appearance or other characteristics of an organism, resulting form the interaction of its genetic constitution with the environment.

Point mutations are changes involving single base pairs.

Recombinant progeny have a different genotype from that of either parent.

Regulatory gene codes for an RNA or protein product whose function is to control the expression of other genes.

Reporter gene is a coding unit whose product is easily assayed.

Restriction enzyme recognizes specific short sequences of unmethylated DNA and cleaves (separation) the duplex.

Restriction fragment length polymorphism (RFLP) refers to inherited differences in sites for restriction enzymes that result in differences in the lengths of the fragments produced by cleavage with the relevant restriction enzyme. RFLP are used for genetic mapping to link the genome directly to a conventional genetic marker.

Restriction map is a linear array of sites on DNA cleaved by various restriction enzymes.

Reverese transcription is synthesis of DNA on a template of RNA; accomplished by reverse transcriptase enzyme.

Southern blotting describes the procedure for transferring denatured DNA from an agarose gel to a nitrocellulose filter where it can be hybridized with a complementary nucleic acid.

Splicing junctions are the sequences immediately surrounding the exon-intron (see exon and intron) boundaries.

TAMBIS-transparent access to multiple biological information sources.

Termination codon is one of three triplet sequences that cause termination of protein sysnthesis; they are also called 'nonsense' codons.

Transcription is synthesis of RNA on a DNA template.

Transcription unit is the distance between sites of initiation and termination by RNA polymerase; may include more than one gene.

Translation is a synthesis of protein on the mRNA template.

Appendix D

Test Images

Synthetic Images:

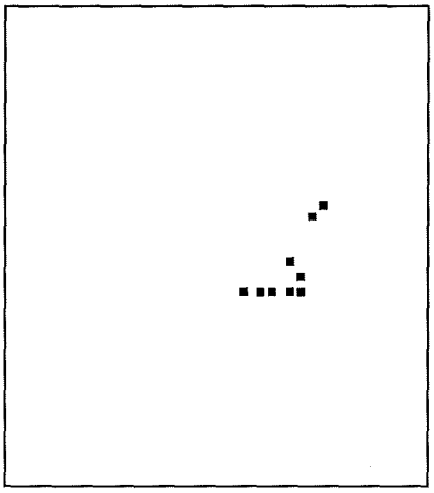


Image: Syn1

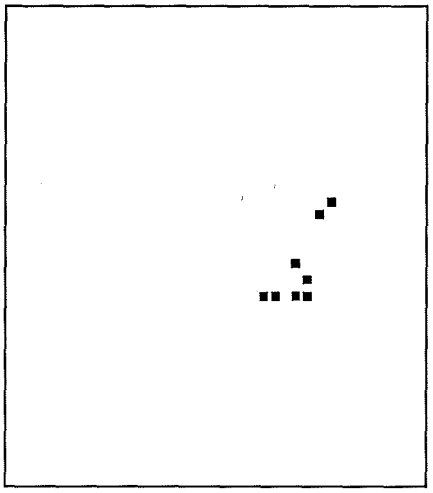


Image: Syn2

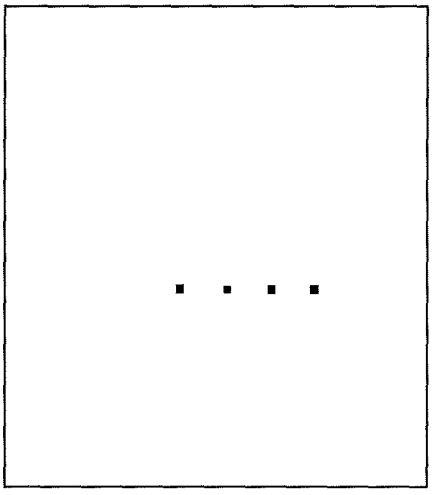


Image: Syn3

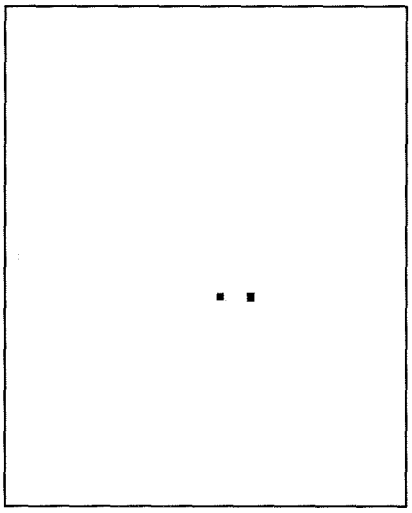


Image: Syn4

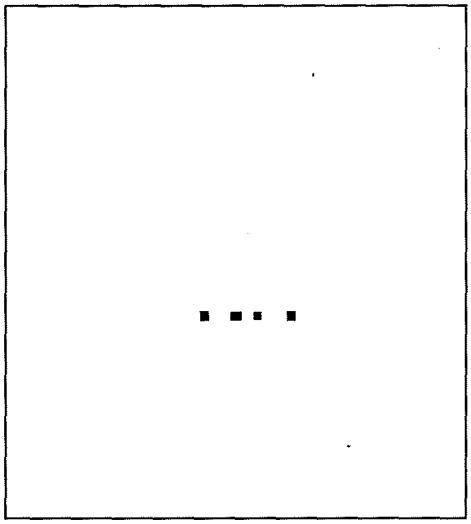


Image: Syn5

Gel Electrophoresis Real Images

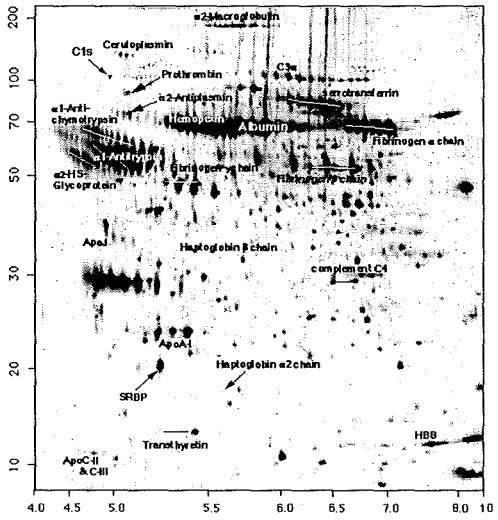


Image: HPG

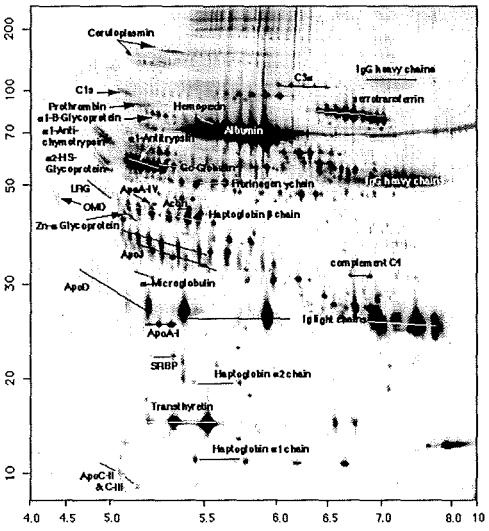


Image: CSF

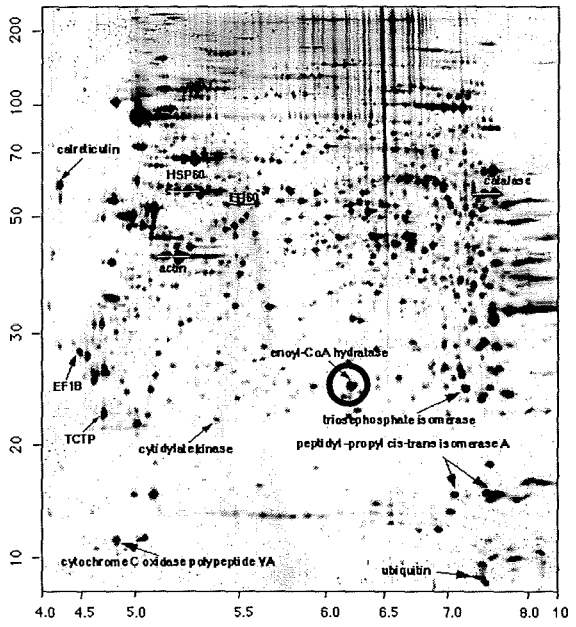


Image: ELC (Source)

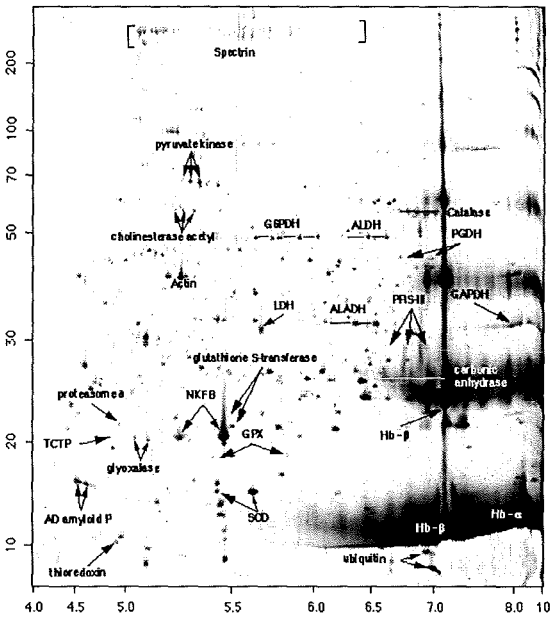


Image: RBC

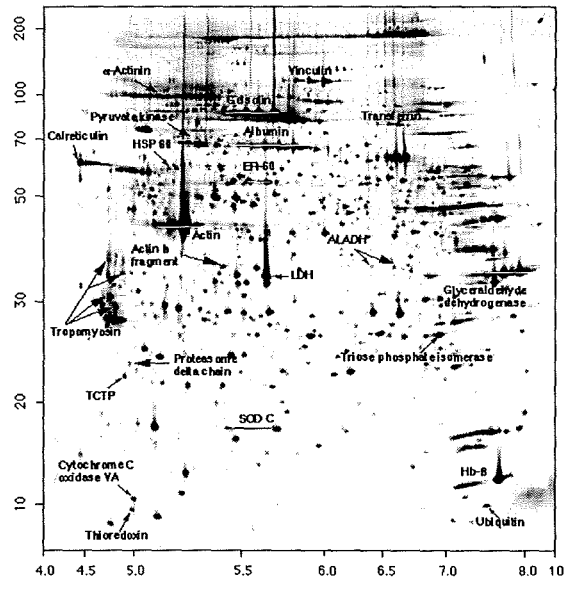


Image: PLT

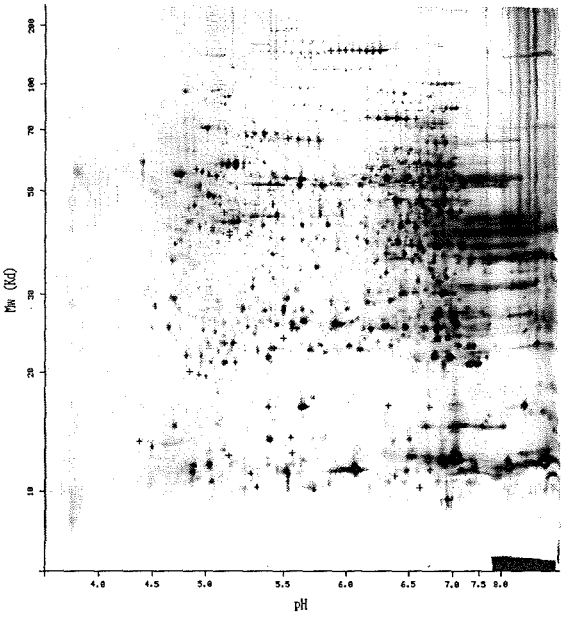


Image: LVR

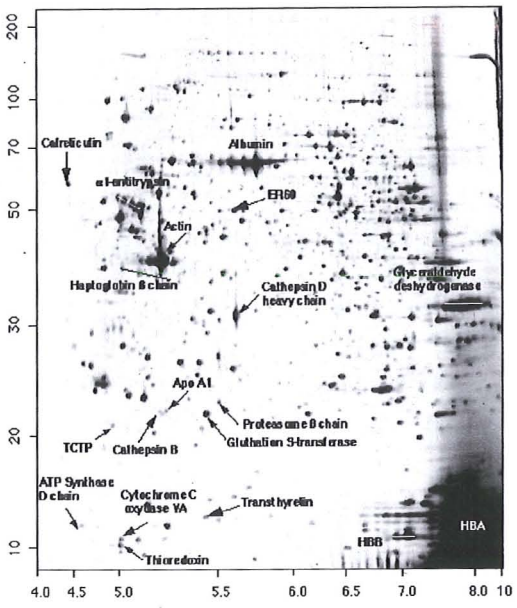


Image: LYP

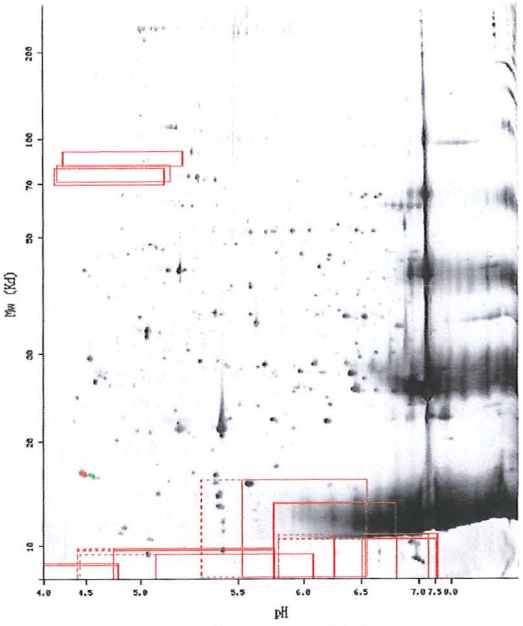


Image: AL4

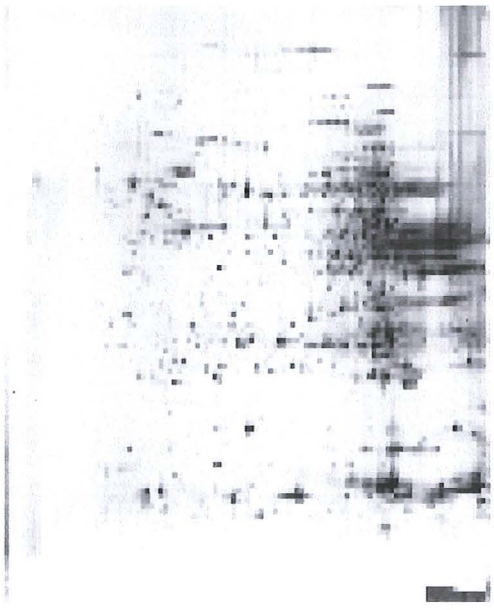


Image: SOD

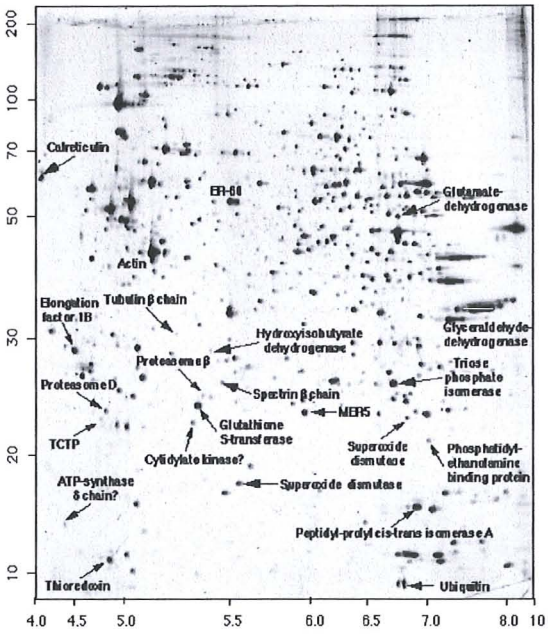


Image: LDL